FORE SYSTEMS®

ForeRunner ATM Services Node 9000 (ASN-9000) Protocol Reference Manual

MANU0273-02 - Rev. A - July 27, 1998

Software Version ASN_FT_5.0.x

FORE Systems, Inc.

1000 FORE Drive
Warrendale, PA 15086-7502
Phone: 412-742-4444
FAX: 412-742-7742

http://www.fore.com

Legal Notices

U.S. Government Restricted Rights. If you are licensing the Software on behalf of the U.S. Government ("Government"), the following provisions apply to you. If the Software is supplied to the Department of Defense ("DoD"), it is classified as "Commercial Computer Software" under paragraph 252.227-7014 of the DoD Supplement to the Federal Acquisition Regulations ("DFARS") (or any successor regulations) and the Government is acquiring only the license rights granted herein (the license rights customarily provided to non-Government users). If the Software is supplied to any unit or agency of the Government other than DoD, it is classified as "Restricted Computer Software" and the Government's rights in the Software are defined in paragraph 52.227-19 of the Federal Acquisition Regulations ("FAR") (or any successor regulations) or, in the cases of NASA, in paragraph 18.52.227-86 of the NASA Supplement to the FAR (or any successor regulations).

Printed in the USA.

This publication is provided by FORE Systems, Inc. "as-is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose. FORE Systems, Inc. shall not be liable for any errors or omissions which may occur in this publication, nor for incidental or consequential damages of any kind resulting from the furnishing, performance, or use of this publication.

Information published here is current or planned as of the date of publication of this document. Because we are improving and adding features to our products continuously, the information in this document is subject to change without notice.

RESTRICTED RIGHTS LEGEND. Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 (October 1988) and FAR 52.227-19 (June 1987).

FCC CLASS A NOTICE

WARNING: Changes or modifications to this unit not expressly approved by the party responsible for compliance could void this user's authority to operate this equipment.

NOTE: The *ForeRunner* ASN-9000 has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15, FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

DOC CLASS A NOTICE

This digital apparatus does not exceed Class A limits for radio noise emission for a digital device as set out in the Radio Interference Regulations of the Canadian Department of Communications.

Le present appareil numerique n'emet pas de bruits radioelectriques depassant les limites applicables aux appareils numeriques de la class A prescrites dans le reglement sur le brouillage radioelectrique edicte par le ministere des Communications du Canada.

VCCI CLASS 1 NOTICE

この装置は、第一種情報処理装置（商工業地域において使用されるべき情報処理装置）で商工業地域での電波障害防止を目的とした情報処理装置等電波障害自主規制協議会(VCCI)基準に適合しております。

従って、住宅地域またはその隣接した地域で使用すると、ラジオ、テレビジョン受信機等に受信障害を与えることがあります。

取扱説明書に従って正しい取り扱いをして下さい。

This equipment is in the Class 1 category (Information Technology Equipment to be used in commercial and/or industrial areas) and conforms to the standards set by the Voluntary Control Council For Interference by Information Technology Equipment aimed at preventing radio interference in commercial and/or industrial areas.Consequently, when used in a residential area or in an adjacent area thereto, radio interference may be caused to radios and TV receivers, etc. Read the instructions for correct handling.

***CE** NOTICE*

Marking by the symbol **CE** indicates compliance of this system to the EMC (Electromagnetic Compatibility) directive of the European Community and compliance to the Low Voltage (Safety) Directive. Such marking is indicative that this system meets or exceeds the following technical standards:

- EN 55022 - “Limits and Methods of Measurement of Radio Interference Characteristics of Information Technology Equipment.”
- EN 50082-1 - “Electromagnetic compatibility - Generic immunity standard Part 1: Residential, commercial, and light industry.”

SAFETY CERTIFICATIONS

ETL certified to meet Information Technology Equipment safety standards UL 1950 3rd Edition, CSA22.2, No. 950-95, EN 60950 (1992) and IEC 950, 2nd Edition.

CANADIAN IC CS-03 COMPLIANCE STATEMENT

NOTICE: The Industry Canada label identifies certified equipment. This certification means that the equipment meets certain telecommunications network protective, operational and safety requirements. The Industry Canada label does not guarantee the equipment will operate to the user's satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local telecommunications company. The equipment must also be installed using an acceptable method of connection. In some cases, the company's inside wiring associated with a single line individual service may be extended by means of a certified connector assembly (telephone extension cord). The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situations.

Repairs to certified equipment should be made by an authorized Canadian maintenance facility designated by the supplier. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

Users should ensure for their own protection that the electrical ground connections of the power utility, telephone lines and internal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.

Caution: Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.

Table of Contents

List of Figures

List of Tables

Preface

CHAPTER 1 Overview

CHAPTER 3 Asynchronous Transfer Mode (ATM)

CHAPTER 6 Internetwork Packet Exchange (IPX)

List of Figures

CHAPTER 1 Overview

CHAPTER 2 AppleTalk (atalk) Subsystem

CHAPTER 3 Asynchronous Transfer Mode (ATM)

List of Tables

CHAPTER 1 **Overview**

CHAPTER 2 **AppleTalk (atalk) Subsystem**

CHAPTER 3 **Asynchronous Transfer Mode (ATM)**

CHAPTER 4 **Digital Network Routing (DECnet)**

CHAPTER 5 **Internet Protocol (IP)**

CHAPTER 6 **Internetwork Packet Exchange (IPX)**

APPENDIX A **Well-Known Ports**

Preface

This manual describes the *ForeRunner* ASN-9000 protocol related subsystems. For information on Boot PROM commands, refer to the *ForeRunner ASN-9000 Installation and Maintenance Manual*. For information on other subsystems, refer to the *ForeRunner ASN-9000 Software Reference Manual*. For information on applying and configuring filters, refer to the *ForeRunner ASN-9000 Filters Reference Manual*.

Chapter Summaries

Chapter 1 - Overview - Provides an overview of the changes in the protocol user interface and added features in support of MPOA on the ASN-9000.

Chapter 2 - AppleTalk (atalk) Subsystem - Describes commands for configuring the ASN-9000 as an AppleTalk router.

Chapter 3 - Asynchronous Transfer Mode (ATM) - Describes commands for configuring and managing the ASN-9000 as an ATM router.

Chapter 4 - Digital Network Routing (DECnet) - Describes commands for configuring and managing the ASN-9000 as a DECnet router.

Chapter 5 - Internet Protocol (IP) - Describes commands for configuring and managing the ASN-9000 as an IP router.

Chapter 6 - Internetwork Packet Exchange (IPX) - Describes commands for configuring and managing the ASN-9000 as an IPX router.

Appendix A - Well-Known Ports - Provides a pointer to RFC 1483, the “Well-known Ports” RFC.

Related Publications

- *ForeRunner ASN-9000 Installation and Maintenance Manual*, MANU0255-02, June 1, 1998
- *ForeRunner ASN-9000 Software Reference Manual*, MANU0272-02, June 1, 1998
- *ForeRunner ASN-9000 Filters Reference Manual*, MANU0280-02, June 1, 1998
- *ForeRunner ASN-9000 Release Notes*, MANU0274--03, June 1, 1998.

Technical Support

In the U.S.A., customers can reach FORE Systems' Technical Assistance Center (TAC) using any one of the following methods:

1. Select the "Support" link from FORE's World Wide Web page:

   http://www.fore.com/

2. Send questions, via e-mail, to:

   support@fore.com

3. Telephone questions to "support" at:

   800-671-FORE (3673) or 724-742-6999

4. FAX questions to "support" at:

   724-742-7900

Technical support for customers outside the United States should be handled through the local distributor or via telephone at the following number:

+1 724-742-6999

No matter which method is used to reach FORE Support, customers should be ready to provide the following:

- A support contract ID number
- The serial number of each product in question
- All relevant information describing the problem or question.

Typographical Styles

Throughout this manual, all specific commands meant to be entered by the user appear on a separate line in bold typeface. In addition, use of the `Enter` or `Return` key is represented as `<ENTER>`. The following example demonstrates this convention:

```
cd /usr <ENTER>
```

File names that appear within the text of this manual are represented in the following style: "...the `fore_install` program installs this distribution."

Command names that appear within the text of this manual are represented in the following style: "...using the **`flush-cache`** command clears the bridge cache."

Subsystem names that appear within the text of this manual are represented in the following style: "...to access the **`bridge`** subsystem..."

Parameter names that appear within the text of this manual are represented in the following style: "...using *`<seg-list>`* allows the segments for which to display specified bridge statistics to be specified."

Any messages that appear on the screen during software installation and network interface administration are shown in `Courier` font to distinguish them from the rest of the text as follows:

```
.... Are all four conditions true?
```

Important Information Indicators

To call attention to safety and otherwise important information that must be reviewed to ensure correct and complete installation, as well as to avoid damage to the FORE Systems product or the system, FORE Systems utilizes the following *WARNING/CAUTION/NOTE* indicators.

WARNING statements contain information that is critical to the safety of the operator and/or the system. Do not proceed beyond a ***WARNING*** statement until the indicated conditions are fully understood or met. This information could prevent serious injury to the operator, damage to the FORE Systems product, the system, or currently loaded software, and is indicated as follows:

WARNING!

Hazardous voltages are present. To reduce the risk of electrical shock and danger to personal health, follow the instructions carefully.

CAUTION statements contain information that is important for proper installation/operation. Compliance with **CAUTION** statements can prevent possible equipment damage and/or loss of data and are indicated as follows:

CAUTION

Damaging to the equipment and/or software could occur if these instructions are not followed.

NOTE statements contain information that has been found important enough to be called to the special attention of the operator and is set off from the text as follows:

NOTE

If the value of the LECS control parameters are changed while the LECS process is running, the new values do not take effect until the LECS process is stopped, and then restarted.

Laser Notice

Class 1 Laser Product:
This product conforms to applicable requirements of 21 CFR 1040 at the date of manufacture.

Class 1 lasers are defined as products which do not permit human access to laser radiation in excess of the accessible limits of Class 1 for applicable wavelengths and durations. These lasers are safe under reasonably foreseeable conditions of operation.

NOTE The Laser Notice section applies only to products or components containing Class 1 lasers.

Safety Precautions

For personal protection, observe the following safety precautions when setting up equipment:

- Follow all warnings and instructions marked on the equipment.
- Ensure that the voltage and frequency of the power source matches the voltage and frequency inscribed on the equipment's electrical rating label.
- Never push objects of any kind through openings in the equipment. Dangerous voltages may be present. Conductive foreign objects could produce a short circuit that could cause fire, electric shock, or damage to the equipment.

Modifications to Equipment

Do not make mechanical or electrical modifications to the equipment. FORE Systems, Inc., is not responsible for regulatory compliance of a modified FORE product.

Placement of a FORE Systems Product

CAUTION

To ensure reliable operation of FORE Systems products and to protect it from overheating, openings in the equipment must not be blocked or covered. A FORE Systems product should never be placed near a radiator or heat register.

Power Cord Connection

WARNING!

FORE Systems products are designed to work with single-phase power systems having a grounded neutral conductor. To reduce the risk of electrical shock, do not plug FORE Systems products into any other type of power system. Contact the facilities manager or a qualified electrician if not sure what type of power is supplied to the building.

WARNING!

FORE Systems products are shipped with a grounding type (3-wire) power cord. To reduce the risk of electric shock, always plug the cord into a grounded power outlet.

Command Syntax

The following expressions are used in this manual when describing command syntax:

AaBbCcDd A term that is being defined. Example:

IP Helper is an enhancement to the **ip** subsystem that allows a system to be boot from a server separated from the boot client by a gateway.

AaBbCcDd A command name. ASN-9000 commands are case-sensitive; they should always be issued in lowercase. Example:

```
dir
```

| 1) Separates the full and terse forms of a command or argument:

- The full form is shown on the left of the **|**.
- The terse form is shown on the right of the **|**.

Example:

```
dir | ls
```

When the command or argument is entered, either the full form or terse form may be used. In this example, either **dir** or **ls** can be used.

2) Separates mutually exclusive command arguments. Example:

```
active-ama|aa cset p[rimary]|b[ackup] <slot>|all
```

In this example, the command **active-ama|aa** can accept either **active-ama** or **aa**, but not both.

[ ] Enclose optional command arguments or options. Example:

```
active-ama|aa [show] [linemode|lm] <slot>|all
```

In this example, the **[ ]** enclose optional arguments. The command can be issued without the argument(s) shown in **[ ]**. However, the argument must be one of the two options listed between the **[ ]**.

<*AaBbCcDd*> Indicates a parameter for which a value is supplied by the operator. When used in command syntax, <*italics*> indicates the value to be supplied. Example:

```
savecfg <filename>
```

In this example, <*filename*> is a parameter for which a value must be supplied with the command when issued.

AaBbCcDd Indicates a field or file name.

An example of a field name is when booting the software, the `login:` prompt is displayed.

A file name example is when booting the software, the system looks for a file name `cfg`.

`AaBbCcDd`
or
`AaBbCcDd`

Indicates text displayed by the software or input typed at the command prompt. To distinguish typed input from command output, the typed input is shown in bold typeface. Example:

```
22:ASN-9000:system# bootinfo
Thu Aug 7 13:03:38 1997 start
Thu Aug 7 13:03:43 1997 nvram boot order: m
boot device: m
```

In this example, the user enters **`bootinfo`** and the software responds with

```
Thu Aug 7 13:03:38 1997 start
Thu Aug 7 13:03:43 1997 nvram boot order: m
boot device: m.
```

CHAPTER 1 Overview

This chapter provides an overview of the user interface for protocols supported by *ForeRunner* ASN-9000. The ASN-9000 supports the ATM, IP, IPX, Atalk, and DECnet protocols. *ForeThought* 5.0.x reflects support of Multiple Protocol over ATM (MPOA) and the addition of several configuration commands.

1.1 Feature Enhancements

The following enhancements are supported by *ForeThought* 5.0.x:

- In the ATM protocol, these enhancements include a number of features specific to Multi-Protocol over ATM. MPOA commands are grouped functionally into the following categories: Multi-Protocol Server (MPS) and Next-Hop Server (NHS). The MPS commands function over LAN Emulation (LANE) and the NHS commands function over IP-Over Non-Broadcast Multi-Access (ION-NBMA). MPOA provides MPOA servers (MPSs) and MPOA Clients (MPCs) and defines the protocols required for MPSs and MPCs to communicate.
- IP protocol scalable routing commands are supported on the ASN-9000 platform. These commands replace and/or enhance existing Internet Protocol (IP), Routing Internet Protocol (RIP), and Open Shortest Path First (OSPF) routing protocols, providing a new Virtual Local Area Network (VLAN) interface layer. Refer to the *ForeRunner ASN-9000 Filters Reference Manual* for information on existing filter setups.

All the protocol user interface commands described in this manual supplement procedures covered in the *ForeRunner ASN-9000 Software Reference Manual*, the *ForeRunner ASN-9000 Filters Reference Manual*, and the *ForeRunner ASN-9000 Installation and Maintenance Manual.*

1.2 Local Area Network Emulation (LANE) Enhancements

1.2.1 Distributed LAN Emulation (DLE)

Distributed LAN Emulation (DLE) provides a way to add LAN Emulation (LANE) services and Emulated Local Area Network (ELAN) redundancy into an ATM network. Prior to FT_5.0.x, theASN-9000 used LAN Emulation Client (LEC) failover for redundancy.

With DLE, the LEC is no longer the redundancy point. Each ELAN can have many peers. Each LEC connects to the closest peer. If that service location were to fail, the LEC simply re-registers with the next nearest peer of the ELAN’s services. This removes the need for the LEC failover mechanism and provides for a more robust failover time.

1.2.2 LANE Plug-n-Play

In ForeThought 5.0.x, the ASN-9000 LEC now includes a “hostname” string when talking to the LAN Emulation Client Server (LECS) and registers with the LAN Emulation Server (LES). The “hostname” string is the SYSTEM NAME.

1.2.3 Per-ELAN LECS Address Configuration

This feature allows the ASN-9000 to be configured in a way that allows each individual LEC instance serving a specific ELAN to contact the LECS in a unique fashion for cases where the users may want to force the ASN-9000 to a different LECS or simply reach a common LECS by a different means.

1.2.4 LANE Security/LES Connection Validation

When security is enabled and a LES receives a join request from a LEC, the LES checks with the LECS to verify this is a valid client for the ELAN.

1.2.5 ILMI LECS Discovery

The LANE 1.0 specification defines several ways a LEC is able to contact the LECS to get the LANE service information it needs. One way is via the use of ILMI.

1.3 Scalable Routing Enhancements

1.3.1 Internet Protocol (IP) Virtual LAN (VLAN) Enhancements

As of FT_4.0.x, IP VLANs were first created, then segments assigned to those VLANs. All IP interface assignment and configuration then referenced the VLAN rather than the individual segment.

In FT_5.0.x, the ability to add and remove individual segments to the VLANs without having to delete the VLAN and fully recreate it have been added.

1.3.2 IP Routing Table Size Management

In FT_5.0.x, the initial size of the IP routing table is now 5k routes. The size of the routing table can be increased by adding memory in increments of 1k to 5k routes.

1.3.3 NBMA Interfaces

Some protocols, such as Classical IP (CLIP) do not offer a means for broadcasting. In *ForeThought* 5.0.x support for Non-Broadcast Multiple Access (NBMA) provide the ability to run IP routing protocols over non-broadcast interfaces.

1.3.4 Increased Support for the Number of IP Filters

In FT_5.0.x, the number of IP filters that can be configured has been increased to 256.

1.3.5 Support for Backup IP RIP Route

RIP can hold additional “next best cost” backup route. In the case an active route disappears, a backup route can be used in its place. This feature is useful in cases of redundant links to a particular destination. A failure of an active link results in the backup RIP route becoming active immediately.

1.3.6 Support for RIP and OSPF Neighbors on NBMA Interfaces

RIP or OSPF can be configured on NBMA Interfaces through a list of neighbor routers. This provides enhanced security and the ability to run RIP and OSPF over NBMA.

1.3.7 Support for Displaying External LSDB Advertisements

In FT_4.0.x, there is no way to display the External LSDB (elsdb) advertisements. The “OSPF lsdb” command only showed the network, summary, and router links, but not the external advertisements. The FT_5.0.x release has two commands for displaying the link state database. The lsdb command displays all the “regular” LSAs, as it did in FT_4.0.x, while the elsdb command shows the external LSAs (external to the ASN-9000).

1.3.8 New Stub Area Features

When adding an OSPF stub area, the user can now specify if the summary LSAs are to be flooded into the stub area. The default behavior is not to flood the summary LSAs into the stub area. In the FT_4.0.x code, there is no such option.

1.3.9 Tracing and Debugging Support

Tracing and debug support has been added to support the new routing and Multiple Protocol Over ATM (MPOA) features in FT_5.0.x. It displays information local to the ASN-9000 and allows the level of messages to display be set.

1.3.10 Login Failure Trap

The login failure trap sends SNMP traps whenever a user fails in an attempt to login to theASN-9000. This feature is enabled whenever the Lock Switch, either on the Packet Engine front panel or through setting the Lock Switch Jumper on the Packet Engine, is set to Lock. Whenever a user makes, and fails, four attempts at logging into theASN-9000, a trap can be sent to the local SNMP management station. The trap contains information as to the date and time when the login was attempted, the Internet Protocol (IP) address of the station attempting the login, the login id used, and the reason the login was rejected.

1.4 Network Management

1.4.1 SNMP/MIBS/Traps

The following new MIBs, traps, and general SNMP manageability have been added and can be tested though normal SNMP queries:

- add / delete IP interface
- IP MIB support: Support for IP interface table and IP route table MIBs
- MIB support for standard rip v2 MIB and proprietary RIP MIB
- OSPF standard MIB support
- MIB tables for VLANs
- LANE services & LEC MIBs
- Login failure trap
- MPOA MIBs

1.5 ATM

1.5.1 Fail-Safe LNNI

The Independent NSAP feature of FT_4.0.x has been made available to the DLE service point in FT_5.0.x. Users setting up a DLE service peer on the PowerCell can use an Independent NSAP address for that peer (not to be confused with the DLE anycast address).

1.5.2 UNI 3.0/3.1 Auto-Select

The FT_5.0.x release allows the PowerCell to connect to an ATM switch running either UNI 3.0 or 3.1.

1.5.3 ILMI 3.0/3.1 Auto-Select

Similar to the UNI 3.0/3.1 auto-select, the PowerCell is also able to connect to an ATM switch port running either version of ILMI and establish full ILMI connectivity on that port.

1.5.4 Token Ring LANE Services

FT_5.0.x allows for LANE 1.0 emulated token rings over ATM. While the ASN-9000 does not offer a token ring media interface or a token ring LEC instance, it can house the LANE services for a token ring emulated LAN on the PowerCell 700.

1.5.5 Outbound Telnet

The FT_5.0.x release provides support for outbound Telnet sessions from the ASN-9000. An outbound Telnet session can be invoked from the telnet subsystem. This session can be invoked from a TTY-based user interface or an inbound Telnet session. A maximum of two outbound Telnet sessions can be launched simultaneously from the ASN-9000.

1.6 Configuration Files

Configuration files contain the physical configuration as well as all configured parameters set during the configuration process. To retain changes in the configuration, save them periodically. The following paragraph explains how to save the configuration information. This information can be saved either to an internal device, a Flash Memory/Compact Flash Card or floppy disk, or to a tftpboot server, whether the this be to the default server or to the IP address of an optional tftp host.

1.6.1 Saving the Configuration File

To save the configuration information to a local device (Flash Memory/Floppy Disk), issue the following command from the `system` subsystem:

```
savecfg|svcfg <file or device name>
```

where

<file or device name> Specifies filename or device. . T his can be either **`fm:`** Flash Memory or **fd**: floppy disk. Issuing this command with no options/parameters saves the configuration to the default filename **`cfg`** on the default device.

To save the configuration files to the configured tftpboot server, issue the following command from the **tftp** subsystem:

```
savecfg|svcfg [-h <host>] <remote-file>
```

-h <host> Specifies the IP address of the TFTP server. (The default server is specified with the **`set server`** command. See *ForeRunner ASN-9000 Software Manual* for detailed instructions on configuring a tftpboot server.)

<remote-file> Specifies the name of the configuration file to be saved. The default filename is **`cfg`**.

1.7 On-Line Help

On-line help is available at each level of the user interface from the console, or active Telnet session. Entering a **?** (or **help**) at any subsystem prompt displays a listing of the commands available in that subsystem. Entering **global help** at any subsystem prompt displays a list of **global commands**. These commands are available at any point in the user interface. The following display shows the results of entering **?** (or **help**) and **global help** from the **system** subsystem.

```
52:ASN-9000:system# ?

system subsystem:

baud                                  readcfg|rdcfg
bootinfo|bi                           reboot
card-swap|cs                          savecfg|svcfg
config                                syslocn
convert-config|ccfg                   sysname
date                                  temperature|temp
dcd-detection|dcd                     tty2
ethaddr|ea                            uptime
idprom|idp                            version|ver
mem                                  passwd

type 'global help' for global commands

type 'shex' to show an example of configuration

53:ASN-9000:system# global help

global subsystem:

alias                                 rcprompt
checksum                              readenv|rdenv
copy|cp                               rename|mv
default-device|dd                     rm
dir                                   saveenv|svenv
format|fmt                            show-config-example|shex
help|?                                stty
history|hi                            su
histchars                             subsystems|ss
logout|bye                            timedcmd|tc
ls                                    type|cat
more                                 pnm
unalias

54:ASN-9000:system#
```

1.7.1 Command Syntax Help

Help is also available for any command within each respective subsystem by entering **help <command>**. The help displayed contains the command syntax and a description of the options and parameters that can be used with the respective command. The following display shows the help available for the **elan** command.

```
34:ASN-9000:atm/lane# help elan
elan add <segment> <elan-name>|-auto [la <les-atm-address> | lu <lecs-atm-address>]
elan delete <elan-name>
elan [show] <elan-name>|all
elan set <elan-name>|all arp-aging|aa <time [secs]>
elan set <elan-name>|all bus-rate|br <packets per second>
elan set <elan-name>|all control-timeout|cto <time [secs]>
elan set <elan-name>|all flush-timeout|fto <time [secs]>
elan set <elan-name>|all forward-delay|fd <time [secs]>
elan set <elan-name>|all max-arp-retry|mar <count>
elan set <elan-name>|all vcc-timeout|vto <time [secs]>

        type "help elan <verb> <option>" for help with specific commands.

35:ASN-9000:atm/lane#
```

1.7.2 Extended Command Syntax Help

Extended help is also available for those commands that have extensive syntactical variations. In the case of the **elan** command shown above, additional help is available by entering **help elan <verb> <option>**. The display below shows the results of the additional help available for the **elan add** command.

```
35:ASN-9000:atm/lane# help elan add
elan add <segment> <elan-name>|-auto [la <les-atm-address> | lu <lecs-atm-address>]

       Associates an ELAN to segment with an optional LES ATM address
       or LECS ATM ADDRESS

        If la <les-atm-address> is specified, LECS usage is disabled.
        If la <les-atm-address> is NOT specified, LECS usage is enabled
        and the default LECS address of that slot is used.
        If lu <lecs-atm-address> is specified, LECS usage is enabled
        and the user specified LECS address is used.

36:ASN-9000:atm/lane#
```

Entering **help set** or **help show** at a system prompt displays those commands within the current subsystem, and global commands, that use either the set or show command verb option. In most cases the show command verb is assumed when the command is entered with no parameters. This is denoted by the command verb being enclosed in square brackets (**[ ]**). The following examples show the results of entering each of the above help entries within the `atm/lane` subsystem.

```
57:ASN-9000:atm/lane# help set

Help available for:

    pnm set multi|old
    pnm [show]
    default-device|dd set <device>

You may obtain more detailed help by giving additional parameters

58:ASN-9000:atm/lane# help show

Help available for:

    default-device|dd [show]
    elan [show] <elan-name>|all
    les [show] <les-elan-name>|all <slot>|<all> [advanced]
    lec [show]   <slot>|all
    at [show] elan=<elan-name>|addr=<mac-address>|all
    vt [show] <elan-name>|all
    stats [show] elan <elan-name>|all elan|if|all
    stats [show] les  <service-name>|all <slot>|all

You may obtain more detailed help by giving additional parameters

59:ASN-9000:atm/lane#
```

1.8 Enhancements in User Interface Command Syntax

The New User Interface (NUI) (for software versions PH_7PE_FT_4.0.0, PH_8PE_FT_4.0.0 and above) represents a substantial improvement over the Old User Interface (Old UI) (for software version 7-2.6.6.x or below). The NUI's intuitive nature, organization, and hierarchical structure will help you use the ASN-9000 more easily and efficiently. The following information shows the change in command-line interfaces.

1.8.1 Subsystem Organization

‘

Changes in the NUI offer significant improvements:

- a more intuitive approach to subsystem names
- an organizational breakdown of subsystems that provides you with more specificity
- a hierarchical subsystem structure that more effectively reflects levels of significance

For example, because you are managing the ASN-9000 *system*, the “main” subsystem in the OUI is now the “system” subsystem in the NUI. A parallel improvement can be seen in the “atm,” “IP,” and “IPX” subsystems. These have been broken up to reflect specific protocols within these subsystems.

The following table displays subsystem changes in the NUI:

Table 1.1 - User Interface changes in the subsystems

Old User Interface	New User Interface
atm	atm, atm/clip, atm/clippvc, atm/1483routed, atm/1483bridged, atm/foreip, atm/lane, atm/mps, atm/mpc, atm/nhs
ip	ip, ip/rip, ip/ospf
ipm	ip/mcast
ipx	ipx, ipx/rip, ipx/sap
main	system
mgmt	media
tcpstack	host

1.8.2 The Command-Line Syntax

The basic change in the command-line syntax is the order in which command words are entered. In the old User Interface, the command action verb preceded the object to be modified. The new User Interface recognizes a more intuitive syntactical sequence: first enter the object to be configured, then the action, then the parameters. For example, in the IP subsystem to add a vlan you would enter:

```
vlan add <vlanid><seglist>
```

1.8.3 General On-Line Command Help

Entering **help|?** at the command prompt, displays the commands that can be executed at that level. An example of this is:

```
2:ASN-9000 system# help

system subsystem:

        baud                                  readcfg|rdcfg
        bootinfo|bi                           reboot
        card-swap|cs                          savecfg|svcfg
        config                                syslocn
        convert-config|ccfg                   sysname
        date                                  temperature|temp
        dcd-detection|dcd                     tty2
        ethaddr|ea                            uptime
        idprom|idp                            version|ver
        passwd

               type 'global help' for global commands

               type 'shex' to show an example of configuration

3:ASN-9000:system#
```

As shown in the display, entering **global help** displays the global commands, and entering **shex** provides examples on configuring the ASN-9000. This information is displayed whenever **help|?** is entered from any command prompt. Entering **global help** and **shex**, result in the following displays:

```
236:ASN-9000:system# global help

global subsystem:

        alias                                 rcprompt
        checksum                              readenv|rdenv
        copy|cp                               rename|mv
        default-device|dd                     rm
```

```
       dir                                      saveenv|svenv
       format|fmt                               show-config-example|shex
       help|?                                   stty
       history|hi                               su
       histchars                                subsystems|ss
       logout|bye                               timedcmd|tc
       ls                                       type|cat
       pnm                                      unalias

237:ASN-9000:system# shex

The following shows a short example to configure ip interface

               ip vlan add 200.200.200.200 2.1   (add a vlan on segment 2.1)
               ip it add 200.200.200.200 200.200.200.200  (add an ip interface)
               ip enable   (enable ip forwarding)

The following shows some commands in subsystem "bridge"

               bridge br penable  2.1   ("port" enable bridging on seg. 2.1)
               bridge br pdisable 2.2   ("port" disable bridging on seg. 2.2)
               bridge st enable   (enable spanning tree)
               bridge st disable  (enable spanning tree)

In summary, there may be "enable/disable" and their derives
such as "penable/pdisable", "senable/sdisable", and etc.
to set a particular feature on and off
Use "help [cnps]enable" and "help [cnps]disable" in each
subsystem to see what can be set on/off.
```

1.8.4 Syntax Help

Syntax help can be received by entering a command that requires additional options or arguments and not providing any additional options or arguments, or if incorrect arguments or options. For example, entering **idprom|idp** at the system command prompt without the additional arguments or options displays the following:

```
3:ASN-9000:system# idp
usage:
       idprom|idp [show] <slot number>|all
4:ASN-9000:system#
```

Detailed help on a command can be provided by entering **help|?** ***command.*** Entering the **help| ?** with a command will prompt a statement with the correct way of entering the specific command, a list of parameters that can be set for that command, and a brief description of the what the command configures. Here is an example of the **help** command. The command is issued in the IP subsystem to learn the correct way of entering the configuration command for enabling/disabling forwarding of IP packets.

```
20:ASN-9000:ip# help fps
fwd-pkts-with-srcrt-option|fps enable|disable


        Enable or disable forwarding of IP packets containing either the
        Loose-source-route or the Strict-source-route option.
```

With some commands, there are numerous command verbs available. Such commands as **interface** may contain the **add**, **delete** and **show** verbs. Entering **help|? interface** results in a display similar to the following:

```
242:ASN-9000:ip# ? interface

Help available for:

       it|interface add <vlanid> <ipaddr>[/<prefixlen>|<mask>]
              [ ift[ype] b[c] | n[bma] | [p[top] <nbr_addr>] ]
       it|interface del[ete] [-p] <vlanid> <ipaddr>|all
       it|interface [show] [<disprestrictors>]

You may obtain more detailed help by giving additional parameters

243:ASN-9000:ip#
```

The previous display shows all of the syntax available for use with the **ip interface** command. Additional help on each of the various options can be obtained by entering **help|? interface** ***verb***, as shown in the following example.

```
243:ASN-9000:ip# ? interface add

it|interface add <vlanid> <ipaddr>[/<prefixlen>|<mask>]
       [ ift[ype] b[c] | n[bma] | [p[top] <nbr_addr>] ]
```

```
Add an IP interface to the given vlan. If <mask> is not specified
then "natural" subnet mask (class A, B, or C address mask) for the
IP address is used. Interface type can be one of broadcast, nbma
and ptop. Neighbor address must be specified only for ptop type.
If interface type is not specified, broadcast is assumed by default.
```

1.8.5 Command-Line Interface

The ASN-9000 is managed through a DOS/UNIX-like command-line user interface. Commands can be issued from a management terminal attached to directly through a TTY connection on the PE or indirectly through an in-band TELNET connection. Refer to Chapter 2 of the *ForeRunner ASN-9000 Software Reference Manual* for a discussion of the software subsystems. Refer to the appropriate section of the *ForeRunner ASN-9000 Software Reference Manual* for discussions of the commands available in each subsystem.

1.8.6 Configuration Files

Configuration changes effected through software commands can be preserved by saving the changes in a configuration file. Changes saved to the file name `cfg` are automatically applied and, following a software reboot, provided the `cfg` file is present on the boot source, applied to the new session.

1.8.7 Parameter Files

Commands can be issued to modify parameters that control user sessions. These parameters include scroll control, TELNET control characters, commands aliases, and timed commands. If session parameters are not saved in environment files, these parameters will be lost when the session is closed.

Environment files can be saved so that the same conditions can be made available in another user session. The environment file can then be read (loaded), reinstating the session parameter changes that were stored in the environment file.

If an environment file is saved under the name `root.env`, it is automatically loaded whenever the system is logged into under root status. Likewise, environment files saved under the name `monitor.env` are automatically loaded when logging on with monitor status or if the user level is changed from root to monitor during a session.

1.8.8 Automatic Segment-State Detection

When enabled, Automatic Segment-State Detection senses when a link (or something configured on the link) is “bad” or “down.” When a “bad” or “down” link is detected on a particular port, the state of the segment is reflected in the software’s interface tables. *ForeView* Network Management software allows link types to be enabled or disabled on a particular port. Through *ForeView* the state of the following link types can be learned:

- AUI
- MAU RPTR
- MAU
- BNC
- BNCT
- 10Base-T
- Fiber
- Unknown

NOTE

To disable automatic segment state detection on a UTP port, rename the configuration file to something other than **`cfg`** and then reboot the system.

1.8.9 Segment Statistics

Access method and protocol statistics related to segment and packet activity can be displayed. For example, state-change statistics for individual segments can be displayed to show how many times a particular segment has gone up or down since the software was last booted.

1.8.10 Traffic Monitoring

Port activity can be monitored at regular intervals. For example, statistics of packet activity or packet errors and collisions on a particular port can be monitored and graphed.

1.8.11 Virtual Local Area Networks (VLANs)

A Virtual Local Area Network (VLAN) is a collection of segments that share the same group name or interface address. Layer-2 VLANs are created by creating a bridge group. The software comes with a default bridge group called `default` that contains all installed ASN-9000 segments.

Layer-3 VLANs can be created by assigning the same IP, IPX, or AppleTalk interface address to multiple segments. When the software determines a packet is to be sent to a Layer-3 VLAN assigned to multiple segments, the software forwards a copy of the packet on each segment. From a physical perspective, when this happens, a separate packet is sent to each physical interface. From a logical standpoint, however, the forwarded packet has been forwarded onto its single destination network or subnet, irrespective of how many physical interfaces that network or subnet is configured on.

1.8.12 Bridging and Routing

The `bridge` subsystem contains commands for configuring and managing the ASN-9000 as an IEEE 802.1d bridge. Up to 32 network (bridge) groups can be defined, each containing any subset of ASN-9000 segments.

1.8.12.1 Bridge Table and Cache

The software maintains a bridge table containing the MAC-layer hardware addresses of devices to which the ASN-9000 is able to bridge packets. The software maintains this table by automatically adding new entries and deleting unused entries. In addition, individual entries can be added or removed, including entries that support multi-homed hosts.

Following is an example of a bridge table. Although only a handful of bridge entries are shown in this example, the bridge table usually contains many entries.

```
98:ASN-9000:bridge# bt


Bridging table (aging time = 60 minutes)
Ethernet-address   Seg   Rule  Flags
00-60-08-b0-97-04  2.1   none
00-00-ef-03-9a-b0   --   none  system permanent
08-00-20-7d-e1-7d  2.1   none
.
.
.
00-a0-24-17-3d-9a  2.1   none
00-a0-98-00-09-d3  2.1   none
00-a0-d1-01-ed-7f  2.1   none
ff-ff-ff-ff-ff-ff   --   none  permanent bmcast

Total entries: 97, Learned entries: 95, Permanent Entries: 2
```

In addition to the bridge table, the software maintains a bridge cache of the most recently used source-destination pairs. A source-destination pair contains a packet's source and destination MAC-addresses. The bridge cache provides a fast path for the bridging software and gives an at-a-glance view of current bridging activity. The bridge cache can be displayed to see the source-destination pairs that are frequently used.

1.8.12.2 802.1d

The ASN-9000 can be used "right out of the box" as an 802.1d Bridge. The designation 802.1d refers to the IEEE specification for this type of bridge. For more information regarding 802.1d bridging, refer to Request for Comments (RFCs) 1493 and 1525.

1.8.12.3 Spanning-Tree

The bridge software includes implementation of the 802.1d Spanning-Tree (ST) algorithm. When enabled, the software identifies and "breaks" loops in the network without requiring configuration changes. Commands in the `bridge` subsystem allow fine-tuning of the ST parameters to fit network needs.

1.8.12.4 IP Routing

Commands in the `ip` subsystem allow segments to be configured for IP routing. Using `ip` commands, IP interfaces can be assigned to individual segments. The IP routing software also supports IP VLANs, enabling a single IP subnet that spans multiple segments to be defined. The following subsections describe major features of the `ip` subsystem. Routing Information Protocol (RIP)

The `ip/rip` subsystem commands enable the ASN-9000 to perform IP routing. Using commands in this subsystem, RIP parameters such as `talk` and `listen` can be configured on a segment-by-segment basis. Statistics for RIP packets can also be displayed.

1.8.12.4.1 Open Shortest Path First (OSPF)

The `ip/ospf` subsystem contains commands that can be used to configure the ASN-9000 as an Open Shortest Path First (OSPF) router. OSPF is a routing protocol that enables each participating router to use a topological map of the network to route packets. OSPF routers exchange route information using link state advertisements (LSAs). An LSA is a packet that reports the link state (up or down) of a router's interfaces that are attached to devices in the OSPF network.

1.8.12.5 AppleTalk Routing

The `atalk` subsystem contains commands that can be used to configure ASN-9000 segments for AppleTalk Phase-2 routing. AppleTalk zones and interfaces can be defined as well as commands to ping AppleTalk nodes.

1.8.12.6 IPX Routing

The ASN-9000 can be configured and managed as an IPX router. In addition, the software provides management information on IPX routers and servers through implementation of IPX Routing Information Protocol (RIP) and Service Advertisement Protocol (SAP). RIP or SAP `talk` and `listen` parameters can be enabled selectively on a per-segment basis to control the flow of RIP and SAP updates.

1.8.12.7 DECnet Routing

The **dec** subsystem contains commands for configuring the ASN-9000 to perform DECnet Phase IV routing. Depending on the configuration of the network, the system can be configured to function as a Level-1 or Level-2 router. DECnet statistics for the system (in its capacity as a DECnet node) and for the individual segments configured as DECnet interfaces can also be displayed.

1.8.13 Route Protocol Statistics

The ASN-9000 can gather statistics for the following Internet routing protocols:

- AppleTalk
- Bridge
- DECnet
- IP
- IPM
- IPX
- OSPFv2
- RIP
- SNMP
- TCP
- TCP/IP

CHAPTER 2 AppleTalk (atalk) Subsystem

The AppleTalk subsystem (**atalk**) contains a complete set of AppleTalk Phase-2 commands used with AppleTalk networks and internets. The *ForeRunner*ASN-9000 Switch can be configured to be used as an AppleTalk internet router to perform AppleTalk routing on any or all of its segments. The ASN-9000 can also be configured as a local router or a backbone router, or as any combination of these types of routers. This chapter discusses the use of the commands available from the **atalk** subsystem, with the exception of the filter commands. The AppleTalk filter commands, indicated below in italics, are discussed in detail in the *ForeRunner ASN-9000 Filters Reference Manual.*

2.1 Accessing the Subsystem

To access the atalk subsystem, enter the following command from any runtime command prompt:

atalk

The commands available in the **atalk** subsystem are:

```
40:ASN-9000:atalk# ?

atalk subsystem:

 arp|at                                 ping
 atalk                                  route|rt
 cache                                  stats
 config                                 zone-data-input-filter|zdif
 getmem                                 zone-data-output-filter|zdof
 interface|it                           zone|zt
 nbp-fwd-filter|nff                     zone-pkt-output-filter|zpof
 name|nt

  type 'global help' for global commands

  type 'shex' to show an example of configuration
```

2.2 Getting Started

Perform these steps to set up AppleTalk routing:

1. Enable the AppleTalk subsystem:
   - Allocate memory for AppleTalk routing.
   - Enable AppleTalk routing.
2. Assign AppleTalk zone names to segments.)
3. Assign AppleTalk network (interface) addresses to segments.

2.2.1 Allocating Memory

Before AppleTalk can be used, sufficient dynamic random access memory (DRAM) must be allocated to enable AppleTalk routing.

NOTE

Allocate immediately after booting to ensure that memory is available. Memory cannot be de-allocated. To free allocated memory, make sure the configuration file does not contain a **getmem** command, then reboot the system.

The **getmem** command is used to allocate memory for AppleTalk. The syntax for this command is:

getmem

The system responds with:

```
2:ASN-9000:atalk# getmem
Memory allocated for AppleTalk routing.
3:ASN-9000:atalk#
```

2.2.2 Enable AppleTalk Routing

The **enable atalk** command is used to enable AppleTalk routing. The syntax for this command is:

```
enable|disable
```

where

enable|disable Specifies whether AppleTalk routing is to be enabled or disabled. The default is **disable**.

The following command enables AppleTalk routing:

```
4:ASN-9000:atalk# enable
AppleTalk Routing: Enabled
5:ASN-9000:atalk#
```

2.2.3 Displaying the Current Configuration

The **config** command can be used to verify that memory is allocated and that AppleTalk routing is enabled. The command also displays the aging time for AppleTalk Address Resolution Protocol (AARP) entries (See Section 2.6.2.). The syntax for this command is:

```
config [show]
```

The following example shows that memory has been allocated, AppleTalk routing is enabled and the AARP aging timer is set to 60 minutes.

```
5:ASN-9000:atalk# config

AppleTalk Router: memory available
AppleTalk Routing: Enabled
AARP Aging Timer: 60 minutes
6:ASN-9000:atalk#
```

2.3 Configuring Segments for AppleTalk

Before AppleTalk packets can be routed, zone names and network addresses need to be assigned to one or more network ranges. Use the zone commands and interface commands to configure network ranges for use with AppleTalk networks.

2.3.1 Zone Commands

Zone Information Protocol (ZIP) is used to maintain and exchange between routers a zone table that contains zone names associated with segments. Use the zone commands to add, display, or delete zone names.

2.3.1.1 Adding, Deleting, and Showing Zone Names

The **`zone|zt`** command is used to assign a zone name to a specified network range, show the current zone assignments, and to delete specified zones.

A *zone name* is an alphanumeric string up to 32 characters in length. Different zone names can be assigned to each network range, multiple zone names can be assigned to the same network range, or the same zone name can be assigned to multiple network ranges. Zone names are not required for non-seed segments. Moreover, for non-seed segments, the assigned zone names are not used. Assigned zone names are used for seed segments.

NOTE

A seed router is a router in an AppleTalk network that has the network number or cable range built in to its port descriptor. The seed router defines the network number or cable range for other routers in that network segment and responds to configuration queries from non-seed routers on its connected AppleTalk network, allowing those routers to confirm or modify their configurations accordingly. Each AppleTalk network must have at least one seed router.

A non-seed router is a router in an AppleTalk network that must first obtain and then verify its configuration with a seed router before it can begin operation.

The zone name assigned to a network range is used by the segment when it attempts to come up as a seed segment. Unless a conflict occurs over the use of the segment as a seed segment, the zone name becomes active for that segment.

Blank spaces can be used in zone names. To add a zone name that contains blank(s), use double quotes around the entire zone name, including the blank(s). The syntax for the **`this`** command is:

```
zone|zt add [-d] <net-range> <zone>
zone|zt [show] [-c] [<disprestrict>]
zone|zt delete <net-range> <zone>
```

where

add Specifies that a zone is to be added with the associated parameters.

[-d] Specifies the default zone for this netrange.

<netrange> Optionally specifies a specific range of network addresses.

<zone> Specifies the zone name to assign to the specified netrange. A zone name is a string of 32 characters that are not case sensitive. (For example, the zone names ADMINISTRATION and administration are regarded by AppleTalk as identical.

[show] Displays the active zone names. An asterisk before the zone name indicates the default zone named for this netrange.

[-c] Shows Appletalk zones that have been configured by the **zt add** command on the segment(s)

[disprestrict] Restricts the display to:
[[seg[ment[s]]]=]<seglist>
z[one]=<zone>
n[et[work|range]]=<x>-<y>

delete Deletes the specified zone.

The following example assigns an AppleTalk zone name of Accounting to netrange 113-119.

```
18:ASN-9000:atalk# zone add 113-119 Accounting
Ok
19:ASN-9000:atalk#
```

This example adds a zone name that contains a leading blank. In this example, the zone name also contains an internal blank.

```
17:ASN-9000:atalk# zone add 120-121 " Tony Net"
Ok
18:ASN-9000:atalk#
```

When AppleTalk zone names are displayed, the names that contain blanks are displayed with quotation marks to show the locations of the blanks. The following example shows how zone names that contain blanks are displayed.

```
19:ASN-9000:atalk# zone -c
AppleTalk Zones Available for Configuration

Net-Range       Zones
---------       -----
113-119         Accounting
120-121         " Tony Net"
20:ASN-9000:atalk#
```

When the zone name is displayed in the Chooser on a Macintosh, the blank spaces appear in the zone name but the quotation marks are not displayed.

NOTE

When the **zone delete** command is used to remove a configured zone name, the change is immediately apparent in the Configured-Zone table, but does not affect zone names on interfaces that are currently up. The change can affect an interface if that interface is capable of seeding and the segment on which the interface is defined is brought down and then back up.

2.4 Configuring AppleTalk Interfaces

After zone names are assigned to one or more network ranges, network addresses must be assigned to each of these network ranges. Each network address consists of:

- Network address range.
- Combination of <*net*>.<*node*>.[1]
- Optionally, the default zone name.

2.4.1 Adding an Interface (Network Address)

The **`interface|it`** command is used to add an AppleTalk interface to one or more segments, delete specified AppleTalk interfaces, or display the current AppleTalk interface configuration. A different network address can be assigned to each net-range, or the same network address can be assigned to multiple net-ranges. When the same network address is assigned to more than one net-range, a VLAN is created. A VLAN is a network that spans two or more net-ranges. A VLAN increases the effective bandwidth of an AppleTalk network without creating additional network numbers. The syntax for this command is:

```
interface|it add [-n] <seglist>
interface|it add <seglist> <net>.<node> net[range] <x>-<y>
interface|it add [-h] <seglist> <net>.<node> net[range] <x>-<y>
interface|it del[ete] [-a] <seglist>
interface|it [show] [-c] [<disprestrict>]
```

where

[-n] Specifies a non-AppleTalk passive backbone is to be added. To configure a segment for a non-AppleTalk (backbone) net, specify **-n**, rather than an address range. Do not specify a network address. A backbone net connects routers; nodes are not directly attached to the net

<seglist> Specifies the segment numbers to assign an AppleTalk network address. Individual segments, a range of segments, or **all** segments can be specified. The examples below show the possible commands.

[1] In some books, this combination of net address and node address is called a "port node address," an "AppleTalk protocol address," or a "DDP address," depending upon the context. This manual and other ASN-9000documentation uses the term "network address" to refer to this combination.

NOTE

To configure a segment as a non-seed segment, specify a network address range of 0-0. Do not specify a network address following the address range.

To create multiple non-seeding segments, issue a separate **`interface add`** command for each net. If multiple segments are specified with the same command, a VLAN is created.

<net>.<node> Specifies the network address assigned to the specified segment. The value specified for <*net*> must be within the range specified by <*x*>-<*y*>.

For *<node>* specify a range from 1 through 253.

NOTE

Do not use this argument if configuring a segment as a non-seed segment or for a non-AppleTalk (backbone) net.

Node addresses 254 and 255 are reserved AppleTalk for EtherTalk; do not use these addresses. If use of these addresses is attempted, an error message is displayed.

net[range] Specifies the network range assigned to a specified segment. Specify a range from **1** through **65023**.

<x>-<y> Specifies the network ranges. For example, a network range of 113-119 can be specified.

[-h] Specifies a hard-seed backbone.

NOTE

For the **interface add -h** command, all the required data in the command must be entered.

The following commands each add a single segment, a number of specific segments, and a range of segments:

```
44:ASN-9000:atalk# it add 2.4
Segment 2.4 Range 0-0 DDP Addr 0-0 added
Configured as non-seeding interface


45:ASN-9000:atalk# it add -n 1.4, 1.6, 1.8, 1.10
Segment 1.4, 1.6, 1.8, 1.10 Range 0-0 Addr 0-0 added
Configured as non-AppleTalk (backbone)interface


46:ASN-9000:atalk# it add -n 2.5-2.8
Segment 2.5, 2.6, 2.7, 2.8 Range 0-0 DDP Addr 0.0 added.
Configured as non-AppleTalk (backbone) interface
```

The following example shows the command used to configure segment 3.1 as a non-seed segment. (Note that no network address range or network address is specified.)

```
19:ASN-9000:atalk# interface add -n 3.1
Segment 3.1 Range 0-0 DDP Addr 0.0 added.
Configured as non-AppleTalk (backbone) interface.
30:ASN-9000:atalk#
```

In this example, the network address range 220 through 500 is assigned to segment 2.5. The network address “220.150” indicates the specific AppleTalk node to which segment 2.5 is assigned:

```
18:ASN-9000:atalk# interface add 2.5 220.150 net 220-500
Segment 2.5 Range 220-500 DDP Addr 220.150 added.
Configured as non-seeding interface.
19:ASN-9000:atalk#
```

2.4.2 Displaying Network Address Information

Information about segments assigned to an AppleTalk network address can be displayed using the **interface show** command. The display shows network address range and zone names that are assigned to a set of segments. By default, the entire table is displayed. The syntax for this command is:

interface|it [show] [-c] [<disprestrict>]

<disprestrict> <seglist> Specifies the segment(s) for which to display AppleTalk network addresses.

n[et[work|range]]=<x>-<y> Specifies the network range assigned to a specified segment. Specify a range from 1 through 65023.

z[one]=<zone> Specifies the zone name for which to display network address information.

-a Lists all configured and non-configured segments.

-c Shows configured interface information. Does not show dynamically entered interface information.

-z Shows zones in abbreviated form

Here are some examples of the use of the **`interface show`** command. In the first example, no arguments are used with the command. Network address information is shown for all segments that have AppleTalk interfaces. Only two AppleTalk network addresses are assigned to ASN-9000 segments. More than one zone can be associated with a segment.

```
20:ASN-9000:atalk# it
Seg     DDP-Addr Net-Range Ty      NC    GarnFrom  ZC      Zones
-----   --------- -------------    --    --------- --      -----
2.1     220.150  220-220  ETH     cf              cf      Macintosh
2.2     2.128    2-2      ETH     ga    2.124     ga      Engineering
2.3     13.30    13-13    ETH     dn              dn
2.4     128.65   128-128  ETH     un              un
```

The table displayed by the **`interface show`** command shows the following information:

Seg The Seg column lists the segment numbers.

DDP-Addr The DDP-Addr column lists the net address for each segment to which a net address has been assigned. In this example, segments 2.1 and 2.2 are assigned AppleTalk net addresses.

Net-Range The Net-Range column lists the net address range assigned to each AppleTalk segment.

Ty The Ty column indicates the media type (in this case, “ETH,” or Ethernet).

NC The NC column indicates whether the segment was a seed segment (making the ASN-9000 a seed router) for the ASN-9000 network assigned to the segment, or learned the network information from another router in the net.

The NC column indicates one of four states: config, unconfig, garnrd, or down. The initial state is unconfig. If a segment is the seed segment for a network, config soon appears under the NC column. If the segment is not a seed segment, it instead relies upon another router for seed information. In such a case, when the segment has learned the network address from another router, the state of the segment changes from unconfig to garnrd. If the segment is not configured as a seed segment and there is no other router on the network, the state remains unconfig.

If the state remains `unconfig`, the ASN-9000 is unable to find a seed for the segment. Check the connections joining the segment to the seed router. If the connections are working properly, the problem might be in the seed router itself.

If a segment has been configured but is attached to a router that is not turned on, or if a segment is attached to a working router but the segment has been either disabled or has not been added to a zone, the segment is listed as down.

If the state is `-cfg`, the segment is part of an AppleTalk VLAN and has gone down. The other segments in the VLAN might still be up.

GarnFrom The GarnFrom column indicates the seed router from which the ASN-9000 got its configuration. If the ASN-9000 is functioning as the seed router, the GarnFrom field is blank.

ZC The `ZC` column indicates whether the interface is a seed router for the zone associated with the segment. Possible states are cf, un, ga, or dn. See the descriptions for NC.

Zones The Zone column lists the active zone(s) for the segment.

In the following example, the **`-z`** argument is used to limit the display to entries for the specified segment (in this case, segment 2.2):

```
21:ASN-9000:atalk# it show -z 2.2
Seg   DDP-Addr   Range  Type   NetCfg  GarnFrom  ZoneCfg  Zones
---   --------   -----  ----   ------  --------  -------  ----
2.2    2.128       2-2   ETH    garnr    2.12     garnrd   FORE Sys.
```

NOTE

If the interface table displays zeros under the DDP-Addr and Range columns, or “down” for the NC and ZC columns, the segment may be down. If the segment is up, check if AppleTalk routing is enabled. See Section 2.2.2 for information on enabling AppleTalk routing.

2.4.3 Deleting a Network Address

The **`interface del`** command is used to remove an AppleTalk network address from a ASN-9000 segment:

```
interface|it del[ete] [-a] <seg-list>
```

-a Deletes the AppleTalk network address from all segments.

NOTE

Unless the **`-a`** argument is used, each segment to which a network is assigned must be specified in order to delete a network assigned to multiple segments.

<seg-list> Specifies the segments from which to delete the assigned AppleTalk network address. List individual segments, or specify a range of segments.

NOTE

If an AppleTalk network address is deleted, or the zone name with which the deleted address was associated is changed or deleted, a minimum 15- minute wait following the zone name change is recommended before re-adding the address. This time is needed by the devices in the AppleTalk internet to exchange update information about the network address and zone name changes.

Here is an example of the use of the **`interface del`** command. In this example, the interface table is displayed to show which interfaces are defined then the unwanted interfaces are deleted.

```
22:ASN-9000:atalk# it

Seg    DDP-Addr   Net-Range Ty    NC       GarnFr    ZC       Zones
---    --------  -------   ---   ------   ------    ------   -----
1.4    220.150   220-230   ETH   config   220.15    config   Macintosh
1.5    2.128     2-2       ETH   garnrd   2.12      garnrd   FORE Sys.
1.6    220.150   220-230   ETH   config   220.23    config   Macintosh
1.7    220.150   220-230   ETH   config   220.23    config   Macintosh
1.8    220.150   220-230   ETH   config   220.23    config   Macintosh
1.9
1.10
1.11
1.12

23:ASN-9000:atalk#  it del 1.4
Okay
```

In the example, the network address associated with segment 1.4 is deleted. Because the optional **-a** argument is not used, all the segments with which the network address is associated must be specified.

The following example uses the **`interface delete`** command with the **`-a`** argument to delete the same network address:

```
24:ASN-9000:atalk# interface del -a 1.6
Okay
```

When the **-a** argument is used, the network address is deleted from all segments to which it is assigned. In this example, network address 220.150 associated with segment 1.6 is deleted from segment 1.6 as well as segments 1.4, 1.7, and 1.8.

2.5 Pinging Other Devices

The ping command is used to generate an ICMP echo request to a specified device address to determine if the address is reachable on the network. To ping an address, use the following command syntax:

```
ping [-t <timeout>] [-size <size>] <net>.<node>
```

[-t<timeout>]	Specifies how many seconds the ASN-9000 waits for a response from the specified device. The default is 15 seconds before timing out unless a time out value is specified.
[-size <size>]	Specifies the packet length. Specify any length from 64 to 586 bytes. The default packet size is 64 bytes.
<net>	

2.6 Using the AARP Table

The AARP is used to create and maintain a table of translations between MAC-layer node addresses and AppleTalk node addresses. The AARP table enables the ASN-9000 to look up the MAC-layer address of another device (node, router, and so on) based on the device's AppleTalk address. Your options for the arp command are as follows:

```
arp|at [show] [<disprestrict>]

arp|at clear

arp|at set age <time>

arp|at [show] age
```

Entries in the AARP table facilitate transmission of packets from the ASN-9000 (acting as an AppleTalk router) to the devices for which MAC-layer addresses are listed.

These entries are either static or learned:

Static entry An entry created when a network address is assigned to a segment. Each time a network address is assigned to a segment using the **`interface add`** command, the ASN-9000 automatically makes a corresponding entry in the AARP table. These entries cannot be deleted unless the corresponding network address is deleted.

Learned entry An entry that the software automatically adds to the AARP table when it learns about a node address from another managed ASN-9000 or other AppleTalk router, or learns of the node address directly from one of its own segments. The ASN-9000 deletes learned entries when they are inactive for the *AARP aging time*.

For information on the AARP aging time, see Section 2.6.2. Each entry in the AARP table lists the following information:

- DDP address of the node (also known as AppleTalk node address).
- Type of connection the segment has. There are four types of connections:

Local Indicates a device is directly attached to the segment.

Router Indicates the route was dynamically learned. Also indicates another AppleTalk router.

Bcast Indicates the entry in the AARP table is broadcast to all devices in the network. A broadcast packet is denoted by a node address of **255**.

blank Indicates a learned address, one that is added by the software. Blank entries also indicate that a node, not a router, is attached.

- MAC-layer address.
- Segment to which the node is attached.

2.6.1 Displaying AARP Entries

The **arp show** command is used to display the entries in the AARP table. The syntax for this command is:

arp|at [show] [<disprestrict>]

<disprestrict> <seglist> Specifies the segment(s) for which to show AARP entries.

<net.node> Specifies the network address for which to display AARP entries.

(* for wildcard) Specifies that a wildcard is to be used in place of <net.node>

Here are some examples of the use of the **arp show** command. In the first example, the command is entered without an argument. The table displayed lists all AARP entries, both static entries and learned entries, for this ASN-9000.

```
53:ASN-9000:atalk# arp

   AARP Table:
DDP Address   Type    MAC Address       ARP AGE Segment(s)


2.5           Local   00-00-ef-02-41-50   10    1.2
2.22                  00-00-94-20-5f-82   20    1.2
2.255         BCast   09-00-07-ff-ff-ff   40    1.2
111.1         Local   00-00-ef-02-41-50   10    1.3,1.4
111.22                00-00-94-21-fd-1c   20    1.3
111.56                00-00-94-21-f2-43   20    1.4
111.255       BCast   09-00-07-ff-ff-ff   40    1.3,1.4
5.1           Local   00-00-ef-02-41-50   20    1.5
5.255         BCast   09-00-07-ff-ff-ff   40    1.5
```

A wildcard (*****) can be specified in place of <*net.node*>. In the following example, all DDP addresses with the net-address “2” are displayed.

```
27:ASN-9000:atalk# arp 2.*
  ARP TABLE:
DDP Address  Type     MAC Address        ARP AGE  Segment(s)
-----------  -----    -----------------  ---      ----------
2.5          Local    00-00-ef-02-41-50  10       1.2
2.22                  00-00-94-20-5f-82  20       1.2
2.255        BCast    09-00-07-ff-ff-ff  40       1.2
```

NOTE If the AARP table is blank, AppleTalk routing might not be enabled. Use the **`config show`** command to verify that routing is enabled.

2.6.2 Setting the AARP Aging Time

The ASN-9000 can be configured to maintain the AARP table by specifying the amount of time learned entries can remain inactive in the AARP table before being removed by the software. This time limit is the AARP aging interval and is independent of the aging time for routing table entries. The **`arp set age`** command is used to set the number of minutes a learned AARP entry can be inactive before it is deleted from the AARP table. The syntax for this command is:

```
arp set age|saa <time>
```

<time> Specifies the number of minutes that inactive entries remain in the AARP table. The minimum aging time is **3** minutes.

In the command prompt below, the **`arp set age`** command is used with the <*time*> argument to change the AARP aging time to 30 minutes.

```
28:ASN-9000:atalk# arp set age 240
ARP Age changed to 4 minutes.
```

The aging time has to be entered as an integral number of minutes (i.e. a multiple of sixty seconds). For example, 3 minutes could be entered either as **arp set age 180** or as **arp set age 3:00**.

2.6.3 Clearing the AARP Table

The **`arp clear`** command is used to clear all learned entries from the AARP table. Following is an example of the use of this command:

```
30:ASN-9000:atalk# arp clear
Okay
```

2.7 Displaying Route Information

Each ASN-9000 serving as a router in an AppleTalk internet uses Routing Table Maintenance Protocol (RTMP) to maintain and exchange between routers a table of information about other AppleTalk routes throughout the internet. The **`route show`** command is used to display the AppleTalk route table. For each route, the table lists the following information:

- Destination network address.
- Network address of the next hop (if the route is to another router).
- Segment number associated with the next hop.
- Cost (number of hops, or intermediate routers).
- State (good, suspect, or bad).

Periodically, each AppleTalk router (including other ASN-9000s serving as AppleTalk routers) broadcasts RTMP packets through each of its segments configured for AppleTalk to the other AppleTalk routers and nodes adjacent to it. As a result, each router in an AppleTalk network always has a current list of routes to the other networks. The syntax for this command is:

`route|rt [show] [-c|-r] [<disprestrict>]`

-c|-r Restricts the display to only directly connected entries (**-c**) or RTMP entries (**-r**).

-t Displays total number of entries only.

<disprestrict> <seglist> Specifies the segments for which to display route information.

<net> Specifies the AppleTalk net address for which to display route information.

Here is an example of the use of the **`route show`** command.

```
21:ASN-9000:atalk# rt
Destination Next Hop    Segment    Cost    State
----------- ---------   -----------------------
2.-2                    1.5        0       good
3-3        2.61         1.5        1       suspect
220-220                 1.4        0       good
774-774    2.61         1.5        1       bad
```

In this example, the routes for four destinations are shown:

A Lists the network address range for each route in the routing table.

B The network address of the router at the next hop. When a destination is local to the router, the next hop field contains dashes (----).

C Indicates the segment number through which the route can be reached.

D Indicates how many hops (routers) a packet must pass through to reach the destination.

E Lists the state of the route.

A route can have one of three states: good, suspect, or bad. Approximately every 10 seconds the ASN-9000s an RTMP packet to each adjacent ASN-9000 to inform of active (good) routes. When an RTMP packet is not received within 20 seconds, the status for the routes changes from good to suspect.

After a route becomes suspect, the ASN-9000 waits an additional 20 seconds to receive the status packet. If the packet is received within 20 seconds, the status is changed from suspect to good. If the packet is not received, the status changes from suspect to bad. When a route's status changes to bad, the ASN-9000 waits another 20 seconds for an RTMP packet. If the packet still is not received, the bad route is removed from the routing table.

Here is an example of the display produced if using the **-c** argument, which displays entries only for directly connected networks:

```
32:ASN-9000:atalk# route -c

Destination    Next Hop Segments   Cost  State
2-2            -------- 1.5        0     good
220-220        -------- 1.4        0     good
```

Because the routes listed in this display are for directly connected destinations, no value appears under the Next Hop column for either route.

Here is an example of the display produced using the **-c** argument and specifying a specific segment:

```
33:ASN-9000:atalk# rt -c 1.4

Destination   Next Hop  Segments  Cost    State
220-220       ----      1.4       0       good
```

The argument used to produce this display restricts the information to only those routes that are directly connected and are attached to segment number 1.4.

NOTE If the route table is blank, AppleTalk routing might not be enabled. Use the **config show** command to verify that routing is enabled.

2.8 Using the Route Cache

The AppleTalk route cache shows the most recently used destination networks for each segment. At any time an at-a-glance picture of AppleTalk-routing activity in your network can be displayed using the AppleTalk route cache.

2.8.1 Displaying the Route Cache

The **cache show** command is used to display the AppleTalk route cache. The syntax for this command is:

cache [show] [*<seglist>*]

<seglist> Specifies the segments to display information in the route cache. If no segment is specified, information for all segments is shown.

Here is an example of the display produced by this command:

```
33:ASN-9000:atalk# cache
Port 1.1:  empty
Port 1.2:  111.22,111.56
Port 1.3:  2.22
Port 1.4:  2.22
Port 1.5:  empty
Port 1.6:  empty
```

This command displays specified segments only:

```
8:ASN-9000:atalk# cache 1.2,1.4,1.5
Port 1.2:  111.22,111.56
Port 1.4:  2.22
Port 1.5:  empty
```

NOTE

The contents of the route cache can change quite rapidly. As a result, successive **cache show** commands can give different results.

2.8.2 Flushing the Route Cache

The **cache clear** command removes all entries for all segments from the route cache. After the cache is flushed, new entries are again added, using the cache's "most-recently-used" algorithm. Thus, the **cache clear** command can be used to ensure that all entries displayed by a subsequent **cache show** command are fresh.

2.9 Displaying NBP Information

The ASN-9000 uses Name Binding Protocol (NBP) to associate names with AppleTalk network numbers, node addresses, socket numbers, and other services. With NBP, a meaningful name can be bound to any service in an AppleTalk internet. For example, to bind the name "`Printer1`" to a socket number to which a printer is attached, NBP could be used. For information on using the NBP command, see the *ForeRunner ASN-9000 Filters Reference Manual.*

NOTE

The NBP table maintained by the ASN-9000 lists only the objects registered with the ASN-9000.

For each service registered with the ASN-9000, the NBP table lists the following information:

- Object name.
- Object type.
- Zone in which the object resides.

To display the NBP table, use the **name show** command. Here is an example of the information displayed by this command:

```
34:ASN-9000:atalk# name
Object Name      ObjectType   Zone
PORT_220.150     Router       Macintosh
ASN-9000         Router       FORE Systems
```

A network administrator used AppleTalk NBP to name the two objects (services) "PORT_220.150" and "ASN-9000." Both objects are registered to this ASN-9000 as type "Router." They belong to different zones, "Macintosh" and "FORE Systems," respectively.

2.10 Displaying Statistics

During operation of AppleTalk networks, the ASN-9000 collects statistics for AARP, Datagram Delivery Protocol (DDP), and AppleTalk Echo Protocol (AEP) packets. The **`stats show`** command is used to display statistics for AppleTalk ARP, DDP, or AEP packets. The syntax for this command is:

```
stats arp|ddp|echo [-t]
```

arp|ddp|echo Specifies the type of AppleTalk protocol to display statistics.

-t Displays statistics collected since the most recent switch reset, rather than those collected since the most recent clear (using the **`stats clear`** command).

The types of statistics the ASN-9000 collects and displays depend upon the protocol type. Here is an example of information displayed for the AARP protocol:

```
35:ASN-9000:atalk# stats arp
ARP Statistics:

Requests received:           992
Replies received:            296
Invalid packets received:    0
Requests sent:               79
Replies sent:                0
Add arp entry failed:        0
```

Here is an example of the information displayed for the DDP protocol:

```
36:ASN-9000:atalk# stats ddp
DDP Statistics

Out Requests:                 93734
Out Shorts:                   0
Out Longs:                    93734
In Receives:                  82180
Forward Requests:             63849
In Local Datagrams:           78658
No Proto Handler:             0
Out No Routes:                0
Too Short Errors:             0
Too Long Errors:              0
Broadcast Errors:             0
Short DDP Errors:             0
Hop Count Errors:             0
Checksum Errors:              0
Config Address Errors:        0
Local Range Conflicts         0
Config Zone Errors:           0
Memory Allocation Errors      0
```

Here is an example of the information displayed for the AEP (echo) protocol:

```
37:ASN-9000:atalk# stats echo
Echo requests received:       39596
Echo replies received:        0
Echo requests sent:          0
```

NOTE

If a table displayed by the **stats** command contains all zeroes for the statistics amounts, AppleTalk routing might not be enabled. Use the **config show** command to verify that routing is enabled.

2.11 Clearing AppleTalk Statistics

To clear the statistics collected since the most recent clear, use the **`stats clear`** command:

```
stats clear arp|ddp|echo
```

arp|ddp|echo Specifies the type of AppleTalk protocol to clear statistics.

2.12 Testing a Network Address

The **`ping`** command can be used to test the accessibility of and round-trip delay to any AppleTalk node. This command sends an AEP packet to the specified node. The AEP packet contains an instruction to the receiving device to forward the packet back to the sending ASN-9000, thus verifying receipt of the packet. To send an AEP packet, use the following command:

```
ping  [-t <timeout>] [-size <pktsize>] <net>.<node>
```

[-t <timeout>] Optionally specifies the number of seconds the ASN-9000 waits to receive a reply packet from the specified node. The default is **15** seconds.

[-size <pktsize>] If the <*timeout*> argument is used, optionally specifies the size of the echo packet to send to the node. The packet size is measured in bytes. Specify a packet size of 64-586 bytes. The default is **64** bytes.

<net>.<node> Specifies the network node to which to send the test packet.

The following example shows the results of the **`ping`** command when an AEP packet is successfully received by the sending ASN-9000:

```
39:ASN-9000:atalk# ping 220.150
220.150 is alive
```

If the target node to which an AEP packet is sent is not found, or if the timeout expires before the return packet is received, an error message is displayed.

In such a case, check the route table for the network on which the specified target node resides. If the network is listed in the table, check the configuration for the target node to ensure it has learned the current network and zone-related information. If the route table and target node are okay, check the physical connections between the ASN-9000 and the target node.

CHAPTER 3 Asynchronous Transfer Mode (ATM)

This chapter describes the commands in the **`atm`** subsystem and how they can be used to configure and manage the ASN-9000 as an edge device. If specific instructions are required to configure a PowerCell segment to use a particular protocol, refer to the appropriate sections:

- To configure parameters for RFC-1483 Bridged Encapsulation over PVC, see section 3.2.
- To configure parameters for RFC-1483 Routed over PVC, see section 3.3
- To configure LANE 1.0 and 2.0, see section 3.4
- To configure LANE/MPOA, see section 3.6
- To configure NHS, see section 3.7
- To configure FORE IP, see section 3.8.
- To configure CL IP, see section 3.9.
- To configure CL IP PVC, see section 3.10

3.1 Accessing the ATM Subsystem

To access the **atm** subsystem, issue the following command at any runtime prompt:

atm

The commands and subsystems available at this level are:

```
3:ASN-9000:atm# ?
atm subsystem:
active-ama|aa                 >mps
1483bridged                   >nhs
clip                          >mpc
clippvc                       protocol|proto
1483routed                    rate-group|rg
config                        stats
foreip                        vc
lane                          uni
```

3.1.1 ATM Port (PHY) Selection Commands

The commands in this section enable the selection and display of information on the primary and backup port (PHY) on the PowerCell 700.

3.1.1.1 Selecting a Port

The PowerCell software uses the primary port by default. If the link to the primary port fails, the backup port automatically takes over, provided a redundant link has been established to the port switch. In the event that the backup port is in use and fails, the software switches back to the primary port.

After the problem that led to the link failure on the primary port is corrected, manually change back to the primary port again. Manually switching between the backup and primary ports is done using the **aa cset** command.[1]

[1] The port-selection commands contain the word “ama” in reference to the AMA (ATM Media Adapter) on the PowerCell 700. Each ATM port on the PowerCell 700 is provided by an AMA.

NOTE When changing from one port to the other, the connection to the ATM switch (and therefore the ATM network) is temporarily lost while the software switches the connections to the new port.

The syntax for the **`active-ama|aa`** command is:

```
active-ama|aa cset p[rimary]|b[ackup] <slot>|all
active-ama|aa cset linemode|lm <mode> <slot>
active-ama|aa cset linktimer|lt <time_sec> <slot>|all
active-ama|aa [show] [linemode|lm] <slot>|all
active-ama|aa [show] linktimer|lt <slot>|all
```

<slot> Specifies the slot that contains the PowerCell module.

Slots are labeled on the chassis. Slot numbers can also be determined by using the **`system config show`** command.

primary|p | backup|b Selects the port to use. If **`primary`** is specified, the port labeled PRIMARY is used. If **`backup`** is specified, the port labeled BACKUP is used. The default is **`primary`**.

NOTE If the backup port is down and the software attempts to switch to the backup port, the software recognizes that the port is not available and immediately switches back to the primary port. Connection to the ATM switch is temporarily lost then quickly re-established.

The following example selects the primary port. To save the port selection, save the ASN-9000 configuration using the **`system`** or the **`tftp savecfg`** commands.

```
117:ASN-9000 atm# aa cset p 2
```

3.1.1.1.1 Verifying the ATM Port Selection

Use the **`ama show`** command to indicate the port that has been selected for use and the port that actually is in use. The syntax for this command is:

`active-ama|aa [show] [linemode|lm]` *<slot>***`|all`**

show Displays the linemode configured for the ATM card.

linemode|lm Allows you to specify that a linemode is to be configured.

<slot>|all Displays the specific slot you have configured.

Slots are labeled on the chassis. Slot numbers can also be determined using the **`system config show`** command. If **`all`** is specified, the AMA information is shown for all PowerCell modules.

Following is an example of the display produced by this command. In this example, the primary port is both selected and in use.

```
1:ASN-9000:atm# aa show 4

AMA Configurations for Slot 4:
                    Primary             Backup (Installed)
----------------------------------------------------------
User Selected AMA : PRIMARY
Actual In Use AMA : PRIMARY
PHY UTOPIA Level  : 1                    1
PHY UTOPIA Version: 2                    2
PHY Protocol Type : 155M OC3             155M OC3
PHY Media Type    : Multimode Fiber      Multimode Fiber
```

The fields in this display show the following information:

User Selected AMA The port specified for normal operation. Unless the port assignments have been changed, the software uses the primary port by default and the backup port only if the link to the primary port fails.

Actual In Use AMA The port that is being used. If the link to the primary port fails, this field shows that the backup port is in use, even though the primary port was selected for use.

PHY UTOPIA Level The PHY UTOPIA level in use by the PowerCell module and port. UTOPIA is an ATM standard for the communication between the PowerCell module and the PHY (port).

PHY UTOPIA Version The version of the PHY UTOPIA in use by the PowerCell module and the port.

PHY Protocol Type The PHY-layer protocol in use on the port. The protocol must be the following:

155M OC3155 Mb/s using an OC-3 connector.

PHY Media Type The type of cable connecting the port to the ATM switch. The cable type can be one of the following:

Multimode Fiber
Single Mode Fiber
CAT5 UTP

Normally, the primary port is selected by the software and used for ATM traffic between the PowerCell module and the ATM switch. However, if the primary link fails or is changed to the backup port, the PowerCell uses the backup port. The **aa show** command shows that the backup port is in use as follows:

```
2:ASN-9000:atm# aa show 4
AMA Configurations for Slot 4:
                    Primary             Backup (Installed)
---------------------------------------------------------
User Selected AMA : PRIMARY
Actual In Use AMA :                      BACKUP
PHY UTOPIA Level  : 1                    1
PHY UTOPIA Version: 2                    2
PHY Protocol Type : 155M OC3             155M OC3
PHY Media Type    : Multimode Fiber      Multimode Fiber
```

Notice that the `Actual In Use AMA` field lists the backup port in use, even though the primary port is selected. To use the primary port again, correct the problem that caused the primary link to fail, then use the **aa cset** command to select the primary port. (See section 3.1.1.1.)

3.1.1.2 Setting the Backup Linktimer

A default linktimer value is set for the event of a primary port failure. Before the backup port automatically takes over during a primary port failure, the ASN-9000 waits for the pre-set time period to determine whether the primary port recovers. The linktimer value is set with the **`active-ama cset`** command. The syntax for the **`ama cset`** command is as follows:

`active-ama|aa cset linktimer|lt` *`<time_sec> <slot>`***`|all`**

linktimer|lt Specifies the default linktimer setting with the **`cset`** command. The PowerCell 700 waits for the duration of time set in link-down condition before switching to the back-up PHY.

<time_sec> Specifies the amount of time in seconds before the primary port switches to the backup port. The range of value is between **`3`** to **`600`** seconds. The default is **`2`** seconds.

<slot>|all Specifies the slot that contains the PowerCell module.

Slots are labeled on the chassis. Slot numbers can also be determined by using the **`system config show`** command.

Following is an example of this command:

```
117:ASN-9000:atm# aa cset lt 30 2
```

3.1.1.2.1 Verifying the Backup Linktimer

Use the **`ama show`** command to display the linktimer value for backup port switchover:

`active-ama|aa [show] [linktimer|lt]` *<slot>***`|all`**

show Displays the linemode configured for the ATM card.

linktimer|lt Specifies the default linktimer setting with the **`cset`** command. The PowerCell 700 waits for the duration of time set in link-down condition before switching to the back-up PHY.

<slot>|all Displays the specific slot you have configured.

Slots are labeled on the chassis. Slot numbers can also be determined by using the **system config show** command. If **all** is specified, the AMA information is shown for all the PowerCell modules in the chassis.

Following is an example of the display produced by this command. In this example, the primary port is both selected and in use.

```
1:ASN-9000:atm# aa lt 2
Slot 2 linkdown_timer 30
2:ASN-9000:atm#
```

3.1.1.3 Displaying and Clearing Statistics

To display statistics for the active AMA selected port, use the **stats show** command:

stats [show] active-ama|aa *<slot>*|**all**

active-ama|aa Displays the active AMAs on the specified slot.

<slot>|all Displays the specific slot configured for AMAs.

Following is an example of the display produced by this command.

```
1:ASN-9000:atm# stats aa all
PHY Type: OC3
Slot: 1
Multi-errored cell:       0
Path RDI soak:          27ms
B1 block error:           0
B2 block error:           0
B3 block error:           0
B1 coding violation:      0
B2 coding violation:      0
B3 coding violation:      0
```

The fields in this display show the following information:

PHY Type The type of the PHY in use by the PowerCell module and the port.

Slot The port that is actually being used. If the link to the primary port fails, this field shows that the backup port is in use, even though the primary port was selected for use.

Multi-errored cell The ATM cells that are received with multiple-bit errors in the 5-byte ATM header.

Path RDI soak The amount of time that a loss-of-cell-alignment (LOCA) condition must be present before a path RDI condition is sent via the outgoing G1 byte.

B1/B2/B3 block error Displays the total number of frames received with B1, B2, and B3 errors.

B1/B2/B3 coding violation Displays the total number of received B1, B2, and B3 bit-interleaved parity (BIP) bits that are in error.

To clear statistics for the active AMA selected port, use the **`stats clear`** command. The command syntax is as follows:

```
stats clear active-ama|aa <slot>|all
```

The command entered:

```
24:ASN-9000:atm# stats clear aa 1
PHY statistics cleared
```

3.1.2 Segment Configuration Commands

The commands in this section configure rate groups and assign an ATM protocol and rate group to each ATM segment.

3.1.2.1 Configuring a Rate Group

Rate groups enable dividing the complete bandwidth into different groups of usage. Up to 16 different groups can be defined by using the **`cset`** command. To configure a rate group for the PowerCell module, issue the following command.

```
rate-group|rg cset <rate-group> <rate> <slot>
```

<rate-group> Specifies the rate group. Specify a number from **1** through **16**. The default rate group for all segments is **1**.

<rate> Specifies the rate in Mb/s or Kb/s.

- To specify the rate in Kb/s, enter "**k**" or "**K**" after the number. For example, to specify 45000 Kb/s, enter the number as **45000k** or **45000K**.
- To specify the rate in Mb/s, enter "**m**" or "**M**" after the number. For example, to specify 45 Mb/s, enter the number as **45m** or **45M**.

If Kb/s or Mb/s is not specified, the software assumes
Mb/s.

Specify a rate from 1 to 155000 Kb/s for group 1 or a rate from 0 through 155000 Kb/s for groups 2 through 16. The default for rate group 1 is 155000 Kb/s. The default for rate groups 2 through 16 is 0 Kb/s.

When configuring the rate group, specify a bit rate that is equal to or lower than the maximum bit rate supported by the physical interface type of the ATM port. For example, if the ATM port is an OC-3 port, the port can transmit at 155000 Kb/s or lower. If a rate group with a higher bit rate to the port is applied, the port still transmits at 155000 Kb/s or less.

The PowerCell software uses rate group 1 for all ATM signalling and ILMI traffic. Therefore, do not configure rate group 1 for 0 Kb/s unless ATM signalling and ILMI traffic on the PowerCell module is to be eliminated.

NOTE

If rate groups were configured in software versions earlier than 7-2.6.4.0, the ASN-9000 software converts the rates to Kb/s when saving the configuration file. Therefore the rates are not changed, but their representation is changed.

<slot> Specifies the slot that contains the PowerCell module. Slots are labeled on the chassis. Slot numbers can also be determined using the **`system config show`** command.

NOTE

The total for all the rate groups for the PowerCell module must be 155000 Kb/s or less.

Following is an example of the **`rate-group cset`** command.

```
3:ASN-9000:atm# rg cset 4 25000 1
```

3.1.2.2 Displaying Rate Groups

To display configuration information about the rate groups configured for the PowerCell module, issue the following command:

`rate-group|rg [show]` *<slot>*|**`all`**

<slot>|all Specifies the slot that contains the PowerCell 700. Slots are labeled on the chassis. Slot numbers can also be determined by using the **`system config show`** command. If **`all`** is specified, the rate groups for all the PowerCell modules are displayed.

Following is an example of the information displayed by this command. In this example, the rate group configuration for the PowerCell module in slot 4 is displayed.

```
3:ASN-9000:atm# rg 4
Rate Group Settings For Slot: 4
-------------------------------------------------------
Group  1:  100000 Kbps
Group  2:    6000 Kbps
Group  3:    3000 Kbps
Group  4:    4000 Kbps
Group  5:       0 Kbps
Group  6:       0 Kbps
Group  7:       0 Kbps
Group  8:       0 Kbps
Group  9:       0 Kbps
Group 10:       0 Kbps
Group 11:       0 Kbps
Group 12:       0 Kbps
Group 13:    5000 Kbps
Group 14:    6000 Kbps
Group 15:    7000 Kbps
Group 16:    8000 Kbps
Total   :  139000 Kbps
Idle    :   16000 Kbps
```

As shown in this example, the Kb/s allocated to each of the 16 rate groups is listed. Following the listings for the individual rate groups, this display lists the total Kb/s allocated among all 16 rate groups and the amount of idle (unallocated) Kb/s, if any.

3.1.2.3 Configuring ATM Segments

Up to 32 logical segments can be configured on the PowerCell 700. For each segment, the protocol and the rate group used by that segment can be specified. Each segment on the PowerCell module can be configured for only one protocol and one rate group.

To configure the protocol and rate group for a segment on the PowerCell module, issue the following command:

`protocol|proto sset` *<proto>* **`<seglist>|all`**

<proto> Specifies the protocol to be used on a segment. Specify one of the following:

fore-ip | f
lane | l
classical-ip | c
classical-ip-pvc | cp
routed-1483 | r1483
bridge-encap | b
ip-over-nbma | i
None | n

NOTE RFC-1483 Encapsulation, FORE IP, and CLIP are supported only on the PowerCell 700, and only in software versions 7-2.6.4.0 and later.

<segment-list>|all Specifies the PowerCell segment being configured. Specify a single segment number, a comma-separated list of segments, or a hyphen-separated range of segments. If **`all`** is specified, all the segments on all PowerCell modules in the chassis are configured to use the ATM protocol and rate group specified.

The following command enters the classical ip protocol on all segments. The next command shows classical ip configured on segments 1.5-1.7 only.

```
36:ASN-9000:atm# proto sset c all
37:ASN-9000:atm# proto sset c 1.5-1.7
```

To configure the rate group for a segment on the PowerCell module, issue the following command:

sset rate-group|rg 1|2|3|4 *<segment>***|all**

1|2|3|4 Associates a segment with a rate group. Specify a rate group number from **1** through **4**. The default rate group for all segments is **1**.

<segment>|all Specifies the PowerCell segment being configured. Specify a single segment number, a comma-separated list of segments, or a hyphen-separated range of segments. If **all** is specified, all segments on all PowerCell modules in the chassis are configured to use the ATM protocol and rate group specified.

The following command sets the rate-group at 2 for segments 1.5 through 1.10:

```
41:ASN-9000:atm# rg sset 2 1.5-1.10
```

3.1.2.4 Displaying Configuration for ATM Segments

To display configuration information for segments on a PowerCell module, issue the **config** command. This command displays the protocol and rate group configured on the specified segment(s) using the **config** command.

config [show] [s[egments]=]*<seglist>***|slot=***<slot#>***|all**

<seglist> .Specifies an individual segment number, a comma-separated list of segment numbers, or a hyphen-separated range of segment numbers.

<slot#> Specifies the slot number.

Following is an example of the information displayed by this command. In this example, the ATM configuration for segments 1.5-1.7 is displayed.

```
4:ASN-9000:atm# config 1.5-1.7
Segment       Protocol      State       Rate Group
----------     --------    ----------   ----------
1.5           classical-ip Disabled     2
1.6           classical-ip Disabled     2
1.7           classical-ip Disabled     2
```

The fields in this display show the following information:

Segment Lists the ASN-9000 segments specified.

Protocol Lists the ATM protocol assigned to the segment. The protocol can be one of the following:

fore-ip | fip
lane | l
classical-ip | c
classical-ip-pvc | cp
routed-1483 | r1483
bridge-encap | b
ip-over-nbma | i
None | n

State Indicates the state of the protocol. If the protocol is disabled, enable it using the appropriate command:

- To enable the FORE IP protocol, use the `atm/foreip` **`senable`** command.
- To enable the LANE 1.0 protocol, use the `atm/lane` **`elan add`** command.
- To enable the Classical IP protocol, use the `atm/clip` **`senable`** command.
- To enable the Classical IP PVC protocol, use the `atm/clippvc` **`sensable`** command.
- To enable RFC-1483 routed and bridge encapsulation, use the `atm/1483routed or atm/1483bridged` **`senable`** command.

Rate-Group Indicates the rate group assigned to the segment.

3.1.2.5 Displaying VCs for Specified ATM Segments

To display the active virtual circuits (VCs) on specified segments, issue the **`vc show`** command. This command displays the number of VCs configured on the specified segments. The syntax for this command is:

`vc [show]  <seglist>|all`

<seglist>|all Specifies the segment number to display. If **`all`** is specified, the information is listed for all active segments.

The following information is listed:

vc [show] displays all VCs which are active on specified ATM slot(s).

For each LANE-enabled segment:	LEC Control Direct PP SVC LEC Control Distribute PMP SVC LEC Multicast Send PP SVC LEC BUS Multicast Forward PMP SVC All LEC Data Direct PP SVCs
For FORE-IP-enabled segment:	All FOREIP Input and Output SVCs
For each 1483-Bridge-Encapsulation-enabled segment:	Input and output 1483 PVCs
For each CLIP-enabled segment:	SVC to CLIP ARP Server on same LIS All SVCs to remote CLIP clients on same LIS

Following is an example of the information displayed by the **`vc show`** command.

```
7:ASN-9000:atm# vc all
Slot 1 has 18 active VCs
----------------------------------------------------
PVCs
----
Bi-directional:   5   14   15   16
LANE Services
-------------
    Outbound      : 485  489
    Bi-directional: 484  488  491  492
LANE Clients
------------
  Segment : 1.1
    Inbound       : 486  490
    Bi-directional: 483  487
```

3.2 RFC-1483 Bridge Encapsulation over PVC

This section describes how to configure a PowerCell segment for RFC-1483 Bridge Encapsulation. The RFC-1483 Bridge Encapsulation provides a simple mechanism for encapsulating MAC layer frames and using Permanent Virtual Circuits (PVCs). This section shows how a PowerCell module configured for RFC-1483 Bridge Encapsulation fits into the ATM network and describes how to configure a PowerCell for RFC-1483 Encapsulation.

Use RFC-1483 Encapsulation if the ASN-9000 is to be connected to an ATM backbone. RFC-1483 requires fewer configuration steps than LANE 2.0, Classical IP over ATM, and FORE IP.

Note that RFC-1483 Encapsulation does not provide ARP services or broadcast/multicast services. Consequently, if the network is dynamic and requires these services, use LANE 2.0, Classical IP, or FORE IP instead of RFC-1483 Encapsulation.

Routed 1483 Bridge over ATM does not use Signalling.

3.2.1 The PowerCell Module and RFC-1483 Encapsulation

When configuring a PowerCell segment for RFC-1483 Bridge Encapsulation, configure a PVC connection between one ASN-9000 and another that traverses the ATM switch. To configure the PVC, assign an incoming Virtual Circuit ID (VCI) and an outgoing VCI to the segment. The VCs are unidirectional. For each PowerCell segment using RFC-1483 Bridge Encapsulation, configure an incoming VCI and outgoing VCI on the ATM switch and on the PowerCell module to which the ATM switch is attached. The incoming VCI number on the PowerCell segment must match the outgoing VCI on the ATM switch's port. Likewise, the outgoing VCI number on the PowerCell segment must match the incoming VCI on the ATM switch's port. See Figure 3.1 below.

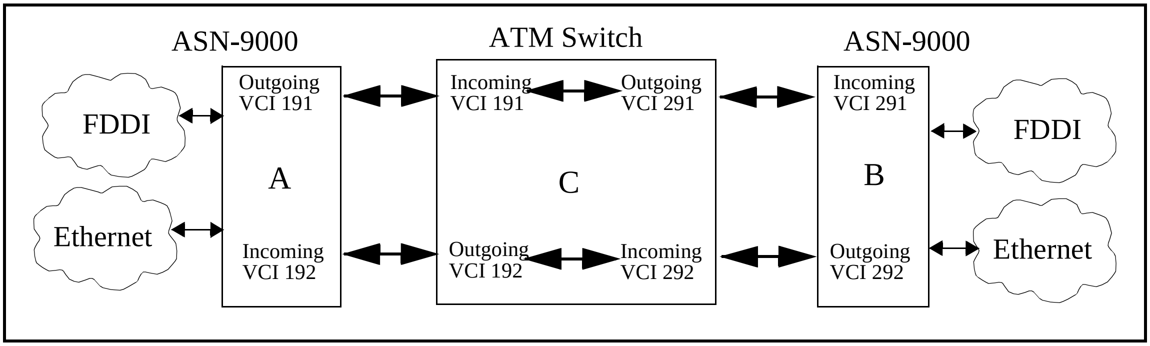


Figure 3.1 - Figure RFC-1483 Bridge Encapsulation and PowerCell Module.

One RFC-1483 Bridge Encapsulation PVC can be configured on each PowerCell segment unless the virtual segment is running another datalink protocol. The PowerCell module supports up to 32 virtual segments, all of which can be allocated to RFC-1483 Encapsulation PVCs if no other datalink protocols are assigned.

3.2.2 Configuring for RFC-1483 Bridge Encapsulation

To use the PowerCell module for RFC-1483 Bridge Encapsulation, perform these steps:

1. Configure to support RFC 1483 Bridge Encapsulation with the **proto sset** command.
2. Assign PVCs to a virtual segment.
3. Enable RFC-1483 Bridge Encapsulation on the segment (**`senable command`**).
4. Verify the RFC-1483 Bridge Encapsulation configuration and operation (**`config show`** command).

3.2.2.1 Configuration Considerations

Before configuring the PowerCell module for RFC-1483 Bridge Encapsulation, make sure the configuration plans are not affected by the following considerations:

- The ASN-9000 implementation of RFC-1483 Bridge Encapsulation allows connections using only Permanent Virtual Circuits (PVCs). Specify the VCIs for a PowerCell segment when configuring the segment for RFC-1483 Bridge Encapsulation.
- VC-based multiplexing is not supported.
- All outgoing packets (packets sent from the PowerCell module to the ATM switch) are sent with the CRC stripped (PID: 0x0007). The ATM switch accepts 0x0001 or 0x0007. The PowerCell sends 0x0007.
- The PowerCell module accepts only PDUs that use Ethernet/802.3 encapsulation. All other PDUs are discarded by the PowerCell module.
- If there are any other protocols configured on the segment, such as FORE IP or LANE, they must first be deleted.

3.2.2.1.1 Configuring the PowerCell

1. Configure the PowerCell segment using the **`atm sset protocol`** command. To configure the PowerCell segment, telnet into or connect to the ASN-9000 through the TTY interface, change to the **`atm`** subsystem, and configure the desired segment.

The following command configures segment 1.2 to use PVC Bridging.

```
17:ASN-9000:atm#proto sset b 1.2
```

2. After configuring bridge encapsulation on a segment, go to the RFC-1483 bridge-encapsulation subsystem to assign PVCs to virtual segments. To get to the RFC-1483 bridge-encapsulation subsystem, issue the following command:

   ```
   atm/1483bridged
   ```

3. Once in the RFC-1483 bridge-encapsulation subsystem, you must assign a PVC to a virtual segment by issuing the following commands:

   ```
   inpvc sset <vci> <seglist>
   outpvc sset <vci> <seglist>
   ```

sset Sets up the incoming or outgoing virtual channel ID on the specified segment.

<vci> Specifies the name of the incoming or outgoing PVC VCI.

<seglist> Specifies the segment on which to send or receive the PVC.

Following is an example of the **`inpvc sset`** command.

```
60:ASN-9000:atm/1483bridged# inpvc sset 45 1.2
```

Following is an example of the outpvc sset command.

```
63:ASN-9000:atm/1483bridged# outpvc sset 46 1.2
```

The **config** command displays the results produced by this command.

```
64:ASN-9000:atm/1483bridged# config 1.2
4:ASN-9000:atm/1483bridged# config 1.2
RFC-1483 encapsulation information for port 1.2
In PVC VCI: 45
Out PVC VCI: 46
Total Pkts sent: 25
Total Pkts rcvd: 322
Pkts rcvd with unknown type: 0
Pkts rcvd with unknown protocol:0
Pkts rcvd with length too big:   0
```

The fields in this display show the following information:

In PVC VCI The incoming PVC’s VCI.

Out PVC VCI The outgoing PVC’s VCI.

Total Pkts sent The number of packets sent on this segment's outgoing PVC. The PowerCell module begins accumulating these statistics when RFC-1483 bridge encapsulation on a segment is enabled.

Total Pkts rcvd The number of packets received on this segment's incoming PVC. The PowerCell module begins accumulating these statistics when RFC-1483 bridge encapsulation on a segment is enabled.

Pkts rcvd with unknown type The number of packets received on this segment's incoming PVC with an unknown type. Any packet that does not contain a SNAP header and an OID of 0080c2 is considered a packet of unknown type.

Pkts rcvd with unknown protocol The PID which is not 0x0001, 0x0007, or 0x000e (STP).

Pkts rcvd with length too big The number of packets received on this segment's PVC with a packet length that is too big. Any packet that exceeds the maximum ethernet packet length of 1518 bytes is considered too big and is dropped.

3.2.2.2 Enabling RFC-1483 Bridge Encapsulation on a Segment

After you have configured the RFC-1483 bridge encapsulation and assigned PVCs to virtual segments, you need to enable this configuration. This is done in the atm/1483bridged subsystem, using the following syntax:

```
senable <seglist>
```

<seglist> Specifies the PowerCell segment on which to enable RFC-1483 bridge encapsulation.

Following is an example of this command. The command enables RFC-1483 bridge encapsulation on segment 1.2 for both incoming and outgoing VCIs.

```
8:ASN-9000:atm/1483bridged# senable 1.2
```

The following command disables the RFC-1483 bridge encapsulation on the segment that was enabled in the previous example:

```
8:ASN-9000:atm/1483bridged# sdisable 1.2
```

3.2.2.3 Verifying PowerCell 1483 Bridge Encapsulation Configuration

The incoming and outgoing VCIs and packet statistics for a PowerCell segment that is enabled for RFC-1483 Bridge Encapsulation can be displayed by issuing the following command:

`config [show]` *`<seglist>`***`|all`**

<segment>|all Specifies the PowerCell segment(s) to display the PVC configuration and statistics. Specify a single segment number or all segments. If **`all`** is specified, PVC information is displayed for all the PowerCell segments in the chassis on which RFC-1483 Bridge Encapsulation is enabled.

Following is an example of this command. In this example, PVC information is displayed for port 8.

```
12:ASN-9000:atm/1483encap# config all
RFC-1483 encapsulation information for port 8
    In PVC VCI:                      182
    Out PVC VCI:                     181
    Total Pkts sent:                 25
    Total Pkts rcvd:                 322
    Pkts rcvd with unknown type:     0
    Pkts rcvd with unknown protocol: 0
    Pkts rcvd with length too big:   0
```

The fields in this display show the following information:

In PVC VCI The incoming PVC’s VCI. Specify this value when configuring a PowerCell segment for RFC 1483 bridged (using the **`inpvc sset`** command).

Out PVC VCI The outgoing PVC’s VCI. Specify this value when configuring a PowerCell segment for RFC 1483 bridged (using the **`outpvc sset`** command).

Total Pkts sent The number of packets sent on this segment’s outgoing PVC. The PowerCell module begins accumulating these statistics when RFC-1483 bridge encapsulation on a segment is enabled.

Total Pkts rcvd The number of packets received on this segment’s incoming PVC. The PowerCell module begins accumulating these statistics when RFC-1483 bridge encapsulation on a segment is enabled.

Pkts rcvd with unknown type The number of packets received on this segment's incoming PVC with an unknown type. Any packet that does not contain a SNAP header and an OID of 0080c2 is considered a packet of unknown type.

Pkts rcvd with unknown protocol The PID which is not 0x0001, 0x0007, or 0x000e (STP).

Pkts rcvd with length too big The number of packets received on this segment's PVC with a packet length that is too big. Any packet that exceeds the maximum ethernet packet length of 1518 bytes is considered too big and is dropped.

After displaying the RFC-1483 Bridge Encapsulation information, verify that RFC-1483 Bridge Encapsulation is operational on the PowerCell module:

1. Check the In PVC VCI and Out PVC VCI fields to make sure they contain the VCIs expected.
2. Place the segment in a live network (if not already done), then re-issue the `config show` command to refresh the display. Check the Total Pkts sent and Total Pkts rcvd fields for signs of activity. If these fields contain zeroes or the other fields indicate errors, do the following:
   a. Reset the ATM module.
   b. Check the RFC-1483 Bridge Encapsulation configuration on the PowerCell module and on the other ATM hardware.
   c. Allow the PowerCell module to operate in the ATM network for a few moments.
   d. Refresh the RFC-1483 Bridge Encapsulation display.
   e. Test the PVC by pinging across the PVC from one endstation to another. The commands used to ping depend upon the type of workstation being used as the endstation.
   f. If the PowerCell module and ATM switch are properly configured but the RFC-1483 Bridge Encapsulation display still shows no packet activity or shows errors, contact FORE Systems TAC.

3.2.2.4 Removing an RFC-1483 Bridge Encapsulation from a Segment

1. Before another protocol on a segment can be configured for RFC-1483 Bridge Encapsulation, the current protocol must be disabled and removed from the segment. If this is not done, the segment is not available to run any other protocol. To disable the segment, issue the **sdisable** command. Following is an example of the command. In this example, the terse form of the command is used.

```
10:ASN-9000:atm/1483encap# sdisable 1.2
```

2. In the above example, RFC-1483 Bridge Encapsulation was disabled on segment 1.2. To verify that the command was successful, issue the **config** command for that segment.
3. After disabling the segment, use the **atm sset protocol none** <*seglist*> command to remove the protocol from the segment. Following is an example of the command. In this example, the terse form of the command is used.

```
10:ASN-9000:atm# sset proto none 1.2
```

4. In the above example, the RFC-1483 Bridge Encapsulation was removed from segment 1.2. To verify that the command was successful, issue the **config** command. The segment is now available and can be configured for use by another protocol.

3.2.2.5 RFC-1483 Bridge Encapsulation Configuration Example

In Figure 3.2, the PVC configuration sets up PVCs from ASN-9000 “A” to ASN-9000 “B” through ATM Switch “C,” and a PVC from “B” to “A” through “C.” Because PVCs are unidirectional, a PVC must also be set in the reverse direction, from ASN-9000 “B” to ASN-9000 “A” through ATM Switch “C.”

Note that each link between the ASN-9000 and the ATM Switch is an independent PVC, and so the same VCI number can be used the full length of the connection as long as the ATM Switch is configured to map the VCIs through the correct ports. In this example, the same VCIs are used in both directions in each PVC to create the connections.

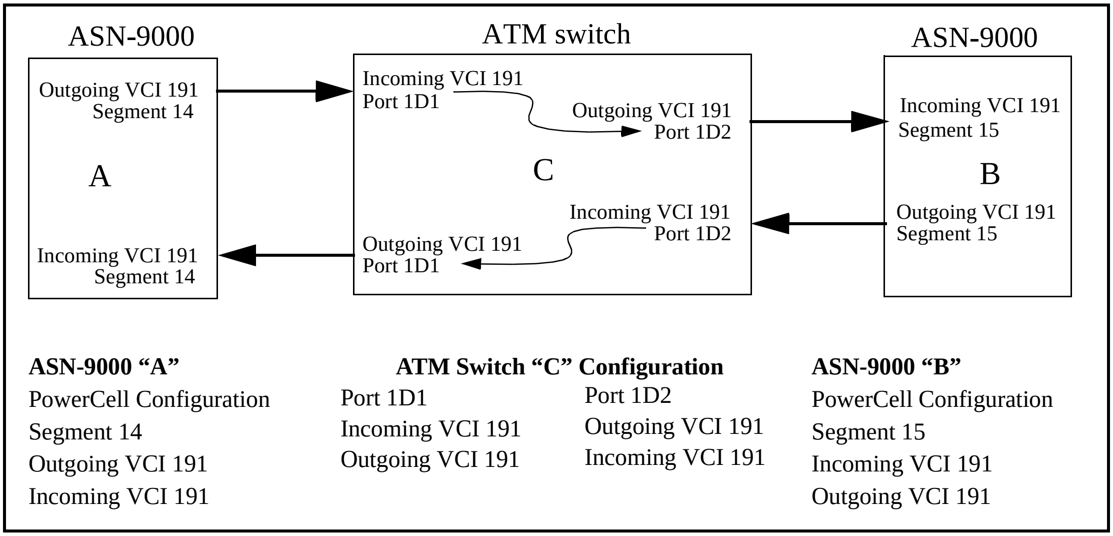


Figure 3.2 - Configuring PVCs from ASN-9000 “A” to ASN-9000 “B”

Note also that all commands must be issued from a management session on the ASN-9000 or switch being configured.

3.2.2.5.1 Selecting VCIs

In Figure 3.3, two different Virtual Channel Identifiers (VCIs) were used for the individual PVCs in each unidirectional connection between the two ASN-9000s. However, the same VCI number can be used along the length of a connection if these VCIs are not being used by other Virtual Channel Connections (VCCs). Using the same VCI for the length of a connection simplifies configuration and management of VCIs, PVCs, and connections. VCI 191 is used for both PVCs from ASN-9000“A” to ASN-9000 “B,” and VCI 192 is used for both PVCs from ASN-9000 “B” back to ASN-9000 “A.”

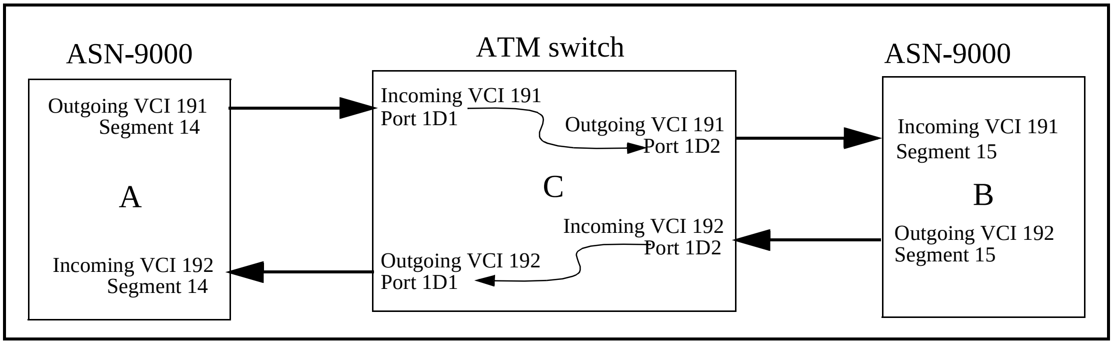


Figure 3.3 - Selecting VCIs

3.2.2.5.2 Examples of PVC Configuration in FORE ATM

Following is an example of the commands issued in the FORE ATM switch to configure the PVCs on the ATM switch to match the configuration shown above. The first command in this example configures the PVC from port 1D1 to port 1D2. The second command sets up the PVC in the reverse direction, from port 1D2 to port 1D1.

NOTE

Do not type the name of the subsystem with the command if already in the subsystem. The example below shows two ways to issue the same command. The first issues the command from within the vcc subsystem, the second from outside the vcc subsystem.

```
localhost::configuration vcc> new 1D1 0 191 1D2 0 191
localhost::configuration> vcc new 1D2 0 192 1D1 0 192
```

3.2.2.5.3 PowerCell Module on ASN-9000 “A”

Following is an example of the ASN-9000 commands issued to configure the segment on the PowerCell module for RFC-1483 Bridge Encapsulation. The `protocol sset` command in the example configures PowerCell segment 1.4 on ASN-9000 “A” to use RFC-1483 Bridge Encapsulation.

```
13:ASN-9000_A:atm# proto sset b 1.4
```

Before segment 1.4 can begin switching traffic through the ATM network, the `protocol sset` command must be issued to enable RFC-1483 Bridge Encapsulation on the segment and configure the incoming and outgoing VCIs. Note that the incoming VCI number on the PowerCell segment must match the outgoing VCI number configured on the ATM switch. Likewise the outgoing VCI number on the PowerCell segment must match the incoming VCI number configured on the ATM switch.

```
14:ASN-9000_A:atm# sset proto b 1.4
```

After configuring the VCIs and enabling the segment for RFC-1483 Encapsulation, verify the configuration and display packet statistics using the **atm/1483bridged: config** command, as shown in the following example.

```
15:ASN-9000_A:atm/1483encap# config 1.4
RFC-1483 encapsulation information for segment 1.4
    In PVC VCI:                      192
    Out PVC VCI:                     191
Total Pkts sent:                 25
    Total Pkts rcvd:                 322
    Pkts rcvd with unknown type:     0
    Pkts rcvd with unknown protocol: 0
    Pkts rcvd with length too big:   0
```

3.2.2.5.4 PowerCell Module on ASN-9000 “B”

Repeat the process used to configure ASN-9000 “A” for the PowerCell Module in ASN-9000 “B.” Use the same commands and use the VCI numbers that match the port numbers configured on the ATM switch that are connected to ASN-9000 “B.” For the example the command would be as follows:

```
13:ASN-9000_B:atm# proto sset b 1.5
```

Then issue the **atm sset protocol** command to enable RFC-1483 Bridge Encapsulation on the segment and configure the VCIs. Note that the incoming VCI number on the PowerCell segment must match the outgoing VCI number configured on the ATM switch. Likewise the outgoing VCI number on the PowerCell segment must match the incoming VCI number configured on the ATM switch.

```
14:ASN-9000_B:atm# proto sset b 1.5
```

After configuring the VCIs and enabling the segment for RFC-1483 Encapsulation, verify the configuration and display packet statistics using the **config show** command as shown in the following example.

```
15:ASN-9000_B:atm# config 1.5
RFC-1483 encapsulation information for segment 1.5
    In PVC VCI:                      191
    Out PVC VCI:                     192
Total Pkts sent:                 34
    Total Pkts rcvd:                 652
    Pkts rcvd with unknown type:     0
    Pkts rcvd with unknown protocol: 0
    Pkts rcvd with length too big:   0
```

3.2.2.6 Spanning-Tree on Bridged 1483

The Spanning-Tree algorithm is a mechanism that logically eliminates physical loops in a bridged 1483 network. For example, PVCs are configured on two different segments through a switch from one PowerCell to another PowerCell, forwarded traffic on one segment loops back to the second segment. When this happens, the network has a loop.

Unless the network topology or the bridges are configured to break the loop, or a mechanism is implemented to logically break the loop, packets are forwarded from bridge to bridge indefinitely, clogging the network. Whenever a segment's state is changed, either by automatic segment-state detection or by a user-interface command, the Spanning-Tree algorithm adjusts the network topology accordingly.

To stop a loop in a 1483 bridged network, enable the Spanning-Tree algorithm using the **`spantree`** command in the `bridge` subsystem. The command below enables the Spanning-Tree algorithm.

```
11:ASN-9000:bridge# st enable
Ok
```

For more information on the Spanning-Tree algorithm command, refer to the *ForeRunner ASN-9000 Software Reference Manual.*

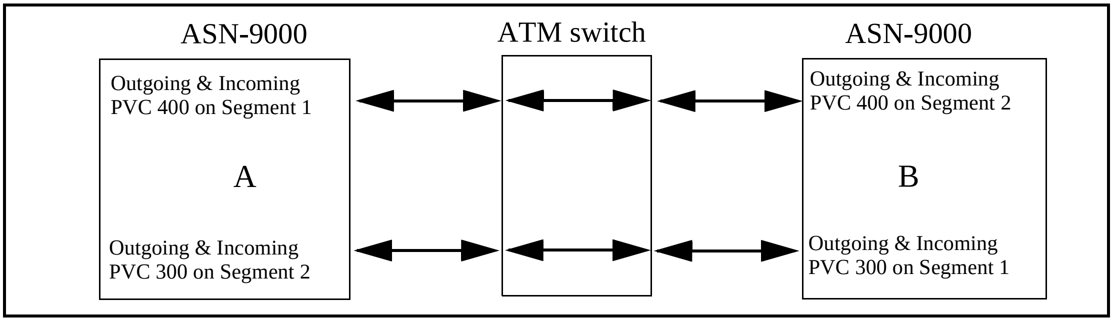


Figure 3.4 - 1483 Bridged Network

Figure 3.4 shows a bridged 1483 network. Segment 1 on ASN-9000 "A" is configured with PVC 400 to segment 2 on ASN-9000 "B." Segment 2 on ASN-9000 "A" is configured with PVC 300 to segment 1 on ASN-9000 "B." The first PVC segment configured is set to a forwarded state for traffic, while the second PVC segment is set to a blocking state eliminating any loopback of packets.

3.3 Routed 1483 over ATM

This section describes the ASN-9000 support for Routed 1483 over ATM. Routed 1483 (RFC 1483) allows transmission of IP datagrams and Address Resolution Protocol (ARP) requests and replies over ATM using ATM Adaptation Layer 5 (AAL5).

NOTE

Routed 1483, unlike Bridged 1483, supports multiple PVCs per virtual interface. Bridged 1483 allows only one PVC. In addition, Routed 1483 performs IP to PVC mapping statically. It does not support Signalling or ARPing of packets.

Normally, ATM connections in a Routed 1483 environment are established dynamically using User Network Interface (UNI) 3.0. ARP, Interim Local Management Interface (ILMI), and UNI 3.0 all work together as when setting up a Switched Virtual Connection (SVC). If a host or switch in a Local IP Subnet (LIS) does not support UNI 3.0, it is not possible to establish a SVC between two hosts. In this case, a Routed 1483 PVC can be used for communication.

On each of the Routed 1483 ASN-9000 segments, the **`sset`** command is used to establish the PVC. An unused VCI must be chosen for each Routed 1483 ASN-9000 segment. PVCs using the chosen VCI must also be setup from each of the hosts to the connecting switch, and then on all of the switches between the two connecting switches.

NOTE

Both the incoming and outgoing connections are set up simultaneously on the host, but they must be set up individually on the switches. The same VCI is used by a host to send on the PVC as well as receive on the PVC. The IP datagrams are sent over the PVC using AAL5 with LLC/SNAP encapsulation.

3.3.1 The PowerCell Module and Routed 1483

Routed 1483 networks contain the following components:

Logical IP Subnet (LIS) A group of IP hosts or routers that are directly attached to an ATM switch and have the same IP network address, subnet address, and subnet mask. One LIS can be configured on each PowerCell segment used for Routed 1483. Individual members of the LIS are joined directly to other members using configured PVCs. Hosts that are not members of the LIS can be reached only by using a LAN router.

The virtual interfaces that are created in Routed 1483 are based on IP and ATM addresses. The interfaces do not use MAC addresses to resolve destinations or routes. Because of this, all packets must be routed when destined for any other interfaces on the ASN-9000, including another LIS on the same PowerCell module.

Figure 3.5 shows an example of an ATM network using Routed 1483. Notice that each ATM host is a member of an LIS. In this example, the hosts are grouped into two LISs: 147.128.10.x with PVCs 100-104 on segment 1 and 147.128.20.x with PVCs 201-205 on segment 2. The subnet mask used in the following example is 255.255.255.0.

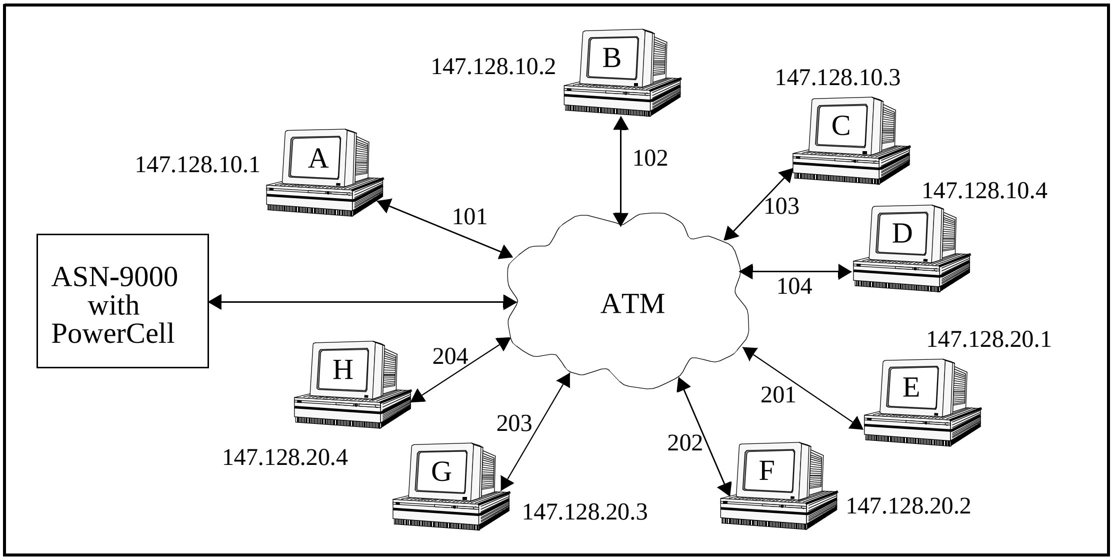


Figure 3.5 - Routed 1483 Network

Figure 3.6 shows stations H, G, F, and E connected to segment 2 using Routed 1483. Configure a PVC to each of these hosts with its associated IP address.

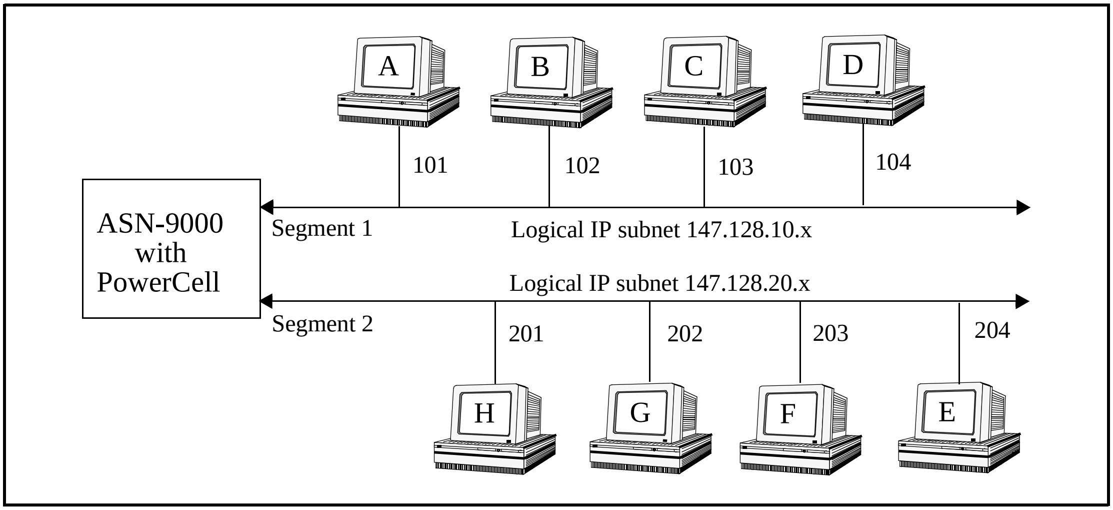


Figure 3.6 - Routed 1483 Network Containing LISs

Figure 3.5 and Figure 3.6 show two LISs connected to a PowerCell module installed in a ASN-9000. Without a router to connect the two LISs, the members of the LISs cannot communicate with each other. The PowerCell module enables the LISs to communicate by routing IP traffic between the LISs. From the ASN-9000 in Figure 3.6, segment 1 connects with PVC 101 to station A, PVC 102 to station B, and so on, with logical IP subnet 147.128.10x.

3.3.1.1 Routed 1483 PVC Support and Packet Encapsulation

The PowerCell module can establish connections between members of an LIS using PVCs. After the PowerCell software establishes a PVC, the software encapsulates IP packets using IEEE 802.2 LLC/SNAP encapsulation, and segments the packets into ATM cells using AAL5.

The default MTU is 9,180 bytes. When the SNAP header is added, the size becomes 9,188 bytes. The maximum packet size is 9180. The same (MTU) size is used for all VCs in a LIS.

3.3.1.2 ATM ARP Support

To configure a PowerCell segment to use Routed 1483, specify the VC and IP address of the destination. (This task is performed using the **`1483routed sset`** command. See section 3.3.2.2.) When a PowerCell segment is configured to support Routed 1483, the segment must be IP configured and routing must be enabled.

3.3.1.2.1 Routed 1483 ARP Display

To display configured IP-PVC pairs for the Routed 1483 segments, use the **`arp show`** command.

`arp show` *`<seglist>`***`|all`**

show Displays the cache entries established.

<seglist>|all Displays the cache entries established on the specified segment.

The following example shows PowerCell segment 5.2 configured for PVC with an IP address of 100.1.1.3. The state of the PVC is “VALID.”

```
117:ASN-9000:atm/1483routed# arp show all
Configured PVCs and state:
IP Address PVC    Segment    State
------------ ---    -------    -----
100.1.1.3       200      5.2      VALID
```

IP Address Indicates the configured destination IP address.

PVC Indicates the configured PVC.

Segment Displays the segment configured for PVC.

State The state always remains “VALID” in Routed 1483.

3.3.1.3 MTU Size

The default MTU size for IP members operating over the Routed 1483 ATM network is 9180 octets. The LLC/SNAP header is 8 octets; therefore, the default Routed 1483 ATM AAL5 protocol data unit size is 9188 octets. In Routed 1483 subnets, values other than the default can be used if all members in the LIS have been configured to use the non-default value.

If a Routed 1483 packet is locally forwarded by the PowerCell module from one LIS to another LIS attached to the same module, the packet is forwarded without being fragmented. However, if the PowerCell module sends the packet to the Packet Engine for processing (for example, if the packet is destined for a segment on another module in the ASN-9000), the module fragments the packet before sending it to the Packet Engine. The fragments can be a maximum of 4060 bytes long.

3.3.2 Configuring a PowerCell Segment for Routed 1483

To use a PowerCell segment for Routed 1483 in an ATM network, perform the following configuration tasks for each segment:

1. Configure an IP interface on the segment (if not already done so). (Use the **`ip interface add`** command.)
2. Set the ATM protocol type to Routed 1483. (Use the **`atm sset protocol`** command.)
3. Enable IP routing.
4. Specify the PVC and the IP address.

The following sections describe how to perform these tasks.

3.3.2.1 Configuration Considerations

Before configuring the PowerCell module for Routed 1483, make sure the configuration plans are not affected by the following considerations:

- Only one IP interface can be configured on a PowerCell segment enabled for Routed 1483.
- Broadcast traffic is not supported, as there is no mechanism in place to distribute broadcast packets. If the segments to be configured require the ability to send and receive broadcast traffic, use LANE on the segments.
- Only one IP interface can be configured on a PowerCell segment, for a maximum of 32 IP interfaces on a PowerCell module.
- Layer-3 VLANs are not supported on PowerCell segments configured for Routed 1483. To configure a Layer-3 VLAN on multiple PowerCell segments, use LANE on the segments.
- Do not include the segments that were configured for Routed 1483 in bridge (network) groups.

3.3.2.2 Configuring a Segment for Routed 1483 on ATM

1. Create a vlan from the `ip` subsystem. The syntax for this command is as follows:

```
vlan add <vlanid> <seglist>
```

```
Below is an example of this command:
2:ASN-9000:ip# vlan add QA 1.8
```

2. Assign the IP address to be used on the segment. (Do this before enabling Routed 1483 on the segment.)

The syntax for this command is:

```
it add <vlanid><ipaddr>
3:ASN-9000: ip#it add QA 12.10.10.3
```

3. Configure the PowerCell segment to use Routed 1483 using the **`protocol sset`** command from the `atm` subsytem. To configure the PowerCell segment to use Routed 1483, telnet into or connect to the ASN-9000 through the TTY interface, change to the `atm` subsystem, and configure the desired segment using the following command:

`proto[col] sset`*<proto>* **`<seglist>|all`**

<proto> Specifies the protocol to be used on a segment. To configure the PowerCell segment to use Routed 1483 issue the following:

`r1483 [routed-1483]`

<segment-list>|all Specifies the PowerCell segment being configured. Specify a single segment number, a comma-separated list of segments, or a hyphen-separated range of segments. If **`all`** is specified, all segments on all the PowerCell modules are configured to use the Routed 1483 protocol.

The example below shows the use of this command:

```
26:ASN-9000:atm# proto sset r1483 1.8
```

4. When setting up the pvc, the destination IP address must be in the same subnet as the one added above to the vlan. From the `atm/1483routed` subsystem, set up the PVC using the following command syntax:

`sset` *`<pvc> <destination-ip-address> <seglist>`*

<pvc> Specifies the PVC to be used on the segment. Valid PVCs range between 32 and 1023.

<destination-ip-address> Specifies the destination IP address of the PVC.

<seglist> Specifies the PowerCell segment being configure

```
10:ASN-9000:atm/1483routed# sset 40 12.10.10.2 1.8
```

5. Enable Routed 1483 on the segment, using the **senable** command. Below is an example of the **senable** command:

```
11:ASN-9000:atm/1483routed# senable 1.8
```

6. Display the 1483routed configuration:

```
78:ASN-9000:atm/1483routed# config 1.8
Configured 1483-ROUTED-PVC's and state
Destination IP Address VC      State   Segment
12.10.10.2             40      enl     1.8
```

3.3.2.3 Removing Routed 1483 from a Segment

To remove Routed 1483 from a segment, perform the following steps:

1. Disable the segment using the **`sdisable`** command (The **`sdisable`** command is described in section 3.3.2.3).
2. Remove one or more configured PVCs from a segment, using the **`pdelete`** command. The **`pdelete`** command is used to delete a single PVCs or all of the PVCs from a Routed 1483 segment.
3. Undefine the protocol using the **`atm proto sset none`** command.

The following example shows how to perform these steps:

```
10:ASN-9000:atm/1483routed# sdisable all
Okay
11:ASN-9000:atm/1483routed# pdelete all
Okay
12:ASN-9000:atm# proto sset None all
Okay
```

3.3.2.3.1 For Members of an LIS

The requirements for IP members (hosts, routers) operating in an ATM LIS configuration are as follows:

- All members have the same IP network number, subnet number, and subnet mask.
- All members within the LIS must be directly connected to the ATM network. Members outside of the LIS can be accessed only by a router .
- All members within the LIS must be able to communicate through ATM with all other members of the same LIS. That is, the VC topology underlying the interconnection among the members must have the ability to be fully meshed.

3.3.3 Displaying the Routed 1483 Configuration

The current Routed 1483 configuration can be displayed using the **`config [show]`** command. Following is an example of the display produced by this command:

```
117:ASN-9000:atm/1483routed# config 1.1
Configured PVCs and state:
PVC     IP Address    State     Segment
---     ----------    -----     -------
300     100.1.1.2     enl       1.1
```

PVC Specifies the PVC associated with the segment.

State Indicates whether Routed 1483 is enabled or disabled on this segment.

Segment Shows the segment running Routed 1483.

3.3.3.1 Displaying and Clearing Statistics

The current Routed 1483 statistics can be displayed using the **`stats [show]`** command. Following is an example of the display produced by this command:

```
117:ASN-9000:atm/1483routed# stats 1.1
Displaying statistics from the ATM Card for segment 1.1
Routed-1483-Over-ATM Statistics for segment 1.1
--------------------------------------------
Connection Fails:                  0
Total Control Packets In:          0
Total Control Packets Out:         0
Arp Replies In:                    0
Arp Replies Out:                   0
Total Arp Replies :                0
Arp Requests In:                   0
Arp Requests Out:                  0
Deleted Arp Replies:               0
Unknown Arp Replies:               0
Total InARP Requests:              0
Total ARP NAKs:                    0
Total bad ARP operations:          0
Total times 1483routed restarted: 1
Unknown Packets received:          0
Unicast Data in:                   17554
Bad ip packets in:                 0
Unicast Packets dropped:           1
Unicast packets forwarded:         17553
```

Use the **`stats clear`** command to clear Routed 1483 over ATM statistics. All learned entries are removed, but static entries (created using the **`sset atmarp`** command) remain in the table. These must be removed manually using the **`pdelete`** command.

This command can be used to help restabilize the network after a host is moved from one segment to another. When there is activity on the network, the cleared entries quickly reappear in the ATM ARP table, and a host that has been moved will be relearned on its new segment.

A Routed 1483 segment is removed on the host side using **`pdelete`** command after disabling the PVC segment using the **`sdisable`** command. Both incoming and outgoing connections are removed simultaneously. The PVC must then be removed from each of the network switches involved.

3.4 LANE Configuration

The ASN-9000 supports the ATM Forum’s Local Area Network Emulation (LANE) 1.0, 2.0, and User-Network Interface (UNI) 3.0, and 3.1 protocol standards. The ATM standards can be used to associate a logical segment of the PowerCell on the ASN-9000 with an Emulated LAN (ELAN). An ELAN is a group of ATM stations that appear to the ASN-9000 as an Ethernet segment (broadcast domain). From the ASN-9000 perspective, ATM stations grouped into an ELAN appear as nodes on a single Ethernet segment. Each ASN-9000 segment can be associated with a separate ELAN.

Lane 1.0 and 2.0 places Ethernet and FDDI LANs on top of an ATM network. The PowerCell module can be used to overlay Ethernet, Fast Ethernet, and FDDI networks managed by the ASN-9000 onto ATM. Each logical segment on the PowerCell module can be associated with one ELAN. The PowerCell can support up to 32 logical segments.

LANE 1.0 and 2.0 emulate the following characteristics of Ethernet LANs:

Connectionless service LANE establishes virtual circuits (VCs) to bridge/route traffic between an Ethernet LAN and ATM, but the VCs are transparent to the Ethernet LAN equipment.

Broadcast and multicast service The Broadcast and Unknown Server (BUS) is a component of LANE that emulates broadcast and multicast services. When the PowerCell module needs to forward broadcast or multicast traffic from an Ethernet network, the module sends the traffic to the BUS, which in turn sends the traffic to each of the destination nodes in the ELAN. The BUS maintains a VC to all LECs participating in the ELAN.

Figure 3.7 shows an example of an ATM network using LANE 1.0. Notice that each ATM station is a member of an ELAN. In Figure 3.7, the stations are grouped into two ELANs: ELAN1 and ELAN2.

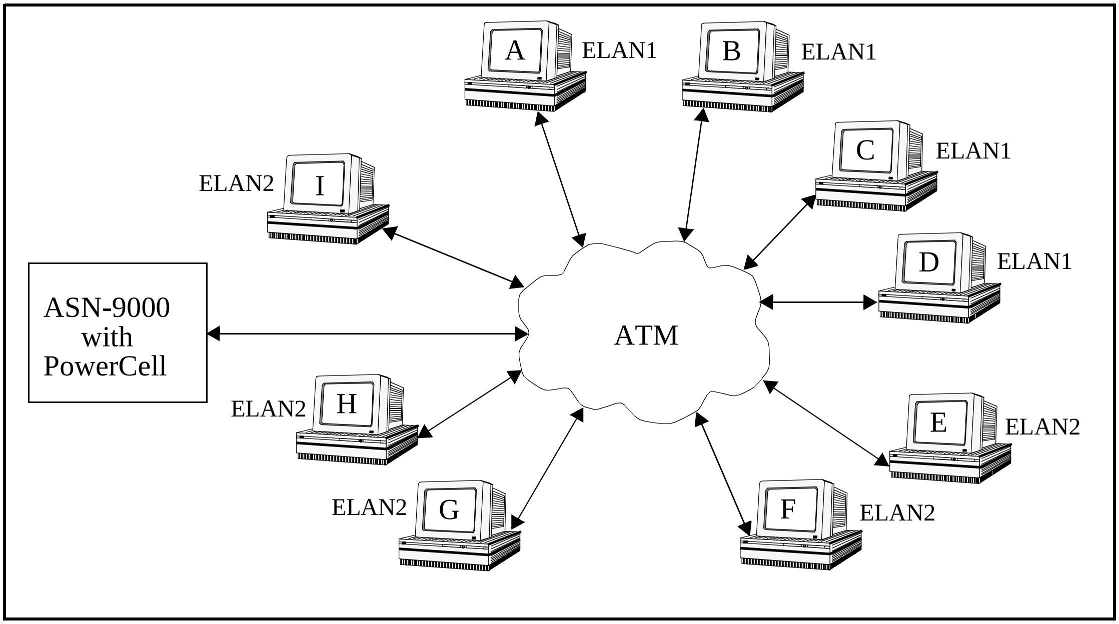


Figure 3.7 - ELANs on the ATM network

Figure 3.8 shows the same ATM LANE 1.0 network from the ASN-9000's perspective. Notice that the ATM stations are still grouped into the same ELANs. However, the ASN-9000 regards each ELAN as an independent Ethernet segment.

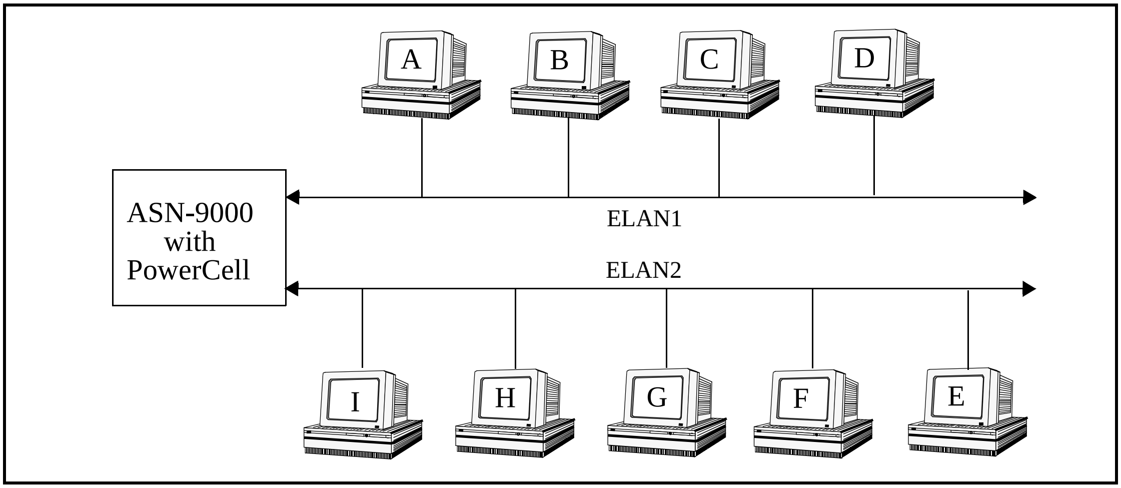


Figure 3.8 - ASN-9000 view of ELANs on the ATM LANE network.

Because ATM stations grouped in an ELAN appear to the ASN-9000 as one a single Ethernet segment, all the configuration features and management features available on the ASN-9000 for Ethernet segments are also available for ELANs. The features include 802.1d bridging (the Spanning-Tree algorithm), IP, IPX, AppleTalk, and DECnet routing protocols, and ARP, as well as automatic segment-state detection, bridge groups, and VLANs.

3.4.1 The PowerCell Module and LANE 1.0 and 2.0

The following sections describe the LANE 1.0 and 2.0 components and the role the PowerCell module plays in a LANE network.

3.4.1.1 LANE Components

LANE 1.0 and 2.0 networks contain four major components:

LAN Emulation Client (LEC)

The LEC is the component in an end system that performs data forwarding, address resolution, and other functions when communicating with other components of an ELAN. A PowerCell segment can be configured as a LEC.

When a PowerCell segment is enabled as a LEC, the PowerCell software performs data forwarding, address resolution, and other control functions when communicating with other components of an ELAN.

LAN Emulation Configuration Server (LECS)

The LECS is responsible for the initial configuration of a LEC. The LECS provides the LEC information about the ELANs that the LEC connects. The LECS also provides the LEC with the LES address (see below) associated with each ELAN.

The LECS can be configured on an ATM switch or a host with an SBA ATM adapter card.

LAN Emulation Server (LES)

The LES is an LAN Emulation ARP (LE_ARP) server that contains address resolution information for an ELAN. The LES contains a table that maps the MAC address of each device in the ELAN to its corresponding ATM address.

The LES can be configured on the ATM switch or on the PowerCell 700.

Broadcast and Unknown Server (BUS)

The BUS emulates the multicast and broadcast functions of an Ethernet segment. When the LEC needs to send a broadcast or multicast packet, or does not know the destination of a unicast packet, the LEC sends the packet to the BUS. The BUS then floods the packet to the appropriate end systems. The BUS can be configured on the ATM switch or on the PowerCell 700. In software version 5.0.x, the LES and BUS are co-located.

3.4.1.2 Advantage of Using a PowerCell Module with LANE 1.0 and 2.0

Figure 3.9 shows an example of an ATM switch connected to multiple LANE ELANs. In this configuration, the ATM switch cannot directly bridge or route traffic from one ELAN to another.

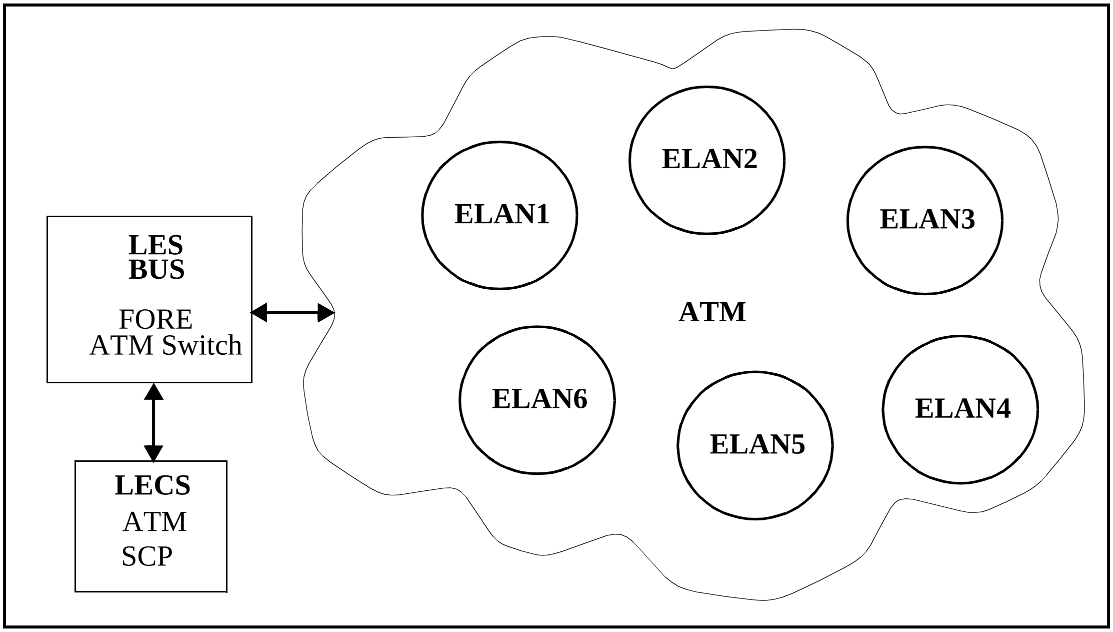


Figure 3.9 - ELANs Configured on the Switch

In Figure 3.9, the ATM switch is switching traffic between LECs on an ELAN. However, the ATM switch cannot bridge or route from one ELAN to another without an ASN-9000. For example, traffic from ELAN1 cannot be bridged or routed to ELAN2. Figure 3.10 shows how adding the PowerCell module to the ATM network enables bridging and routing among LANE ELANs.

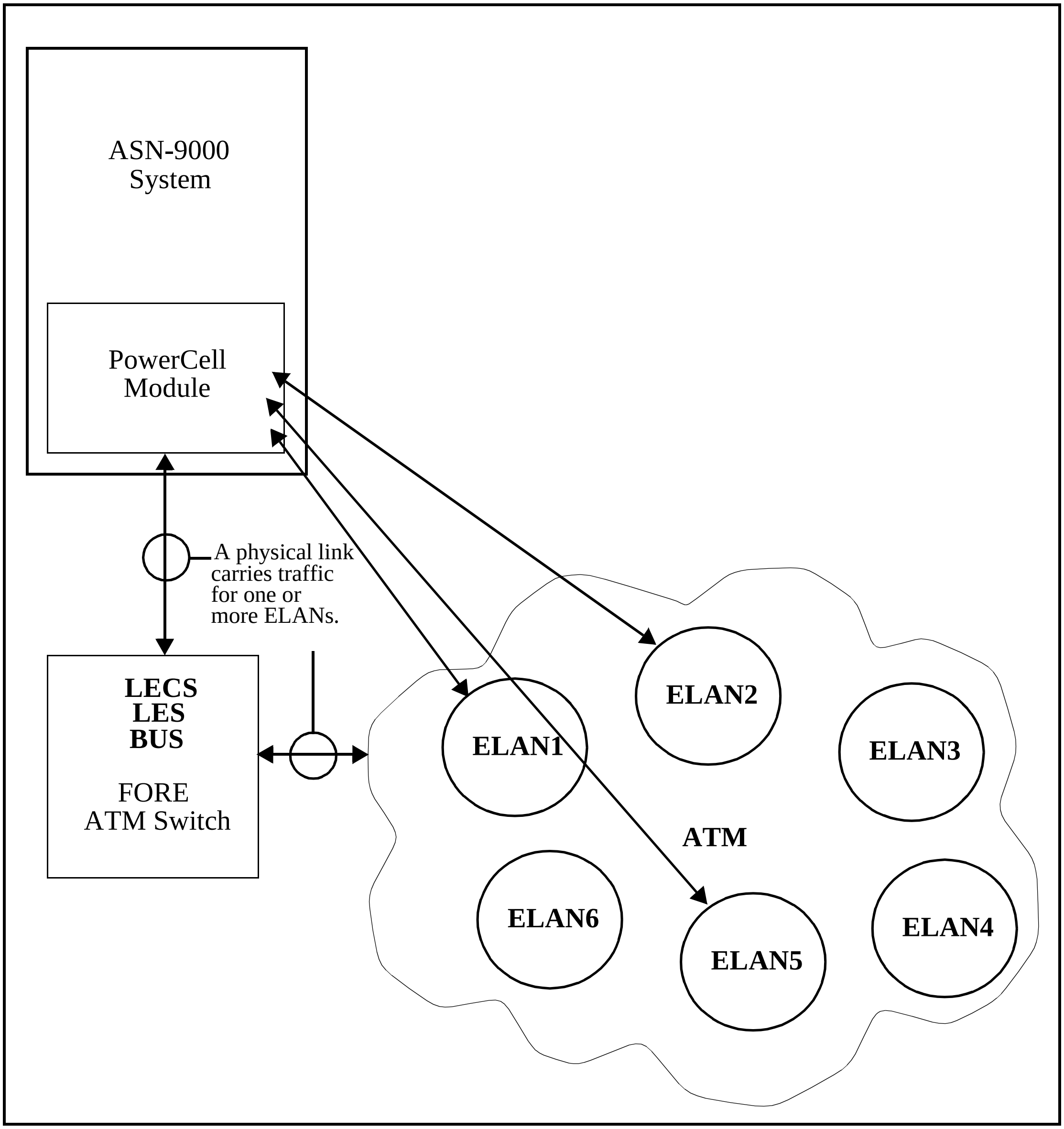


Figure 3.10 - Bridging and Routing Among LANE ELANs

As shown in Figure 3.10, a PowerCell module has been added to the ATM network. LANE traffic from one ELAN to another is sent by the ATM switch to the PowerCell, which uses its on-board ATM processing software to forward the traffic to the appropriate ELAN. For example, traffic sent from ELAN1 to ELAN2 is received by the ATM switch, which sends the traffic to the PowerCell module. The PowerCell module receives the LANE packets, removes the LANE headers, then examines the destination and source addresses of the packet for forwarding information.

3.4.1.3 Token Ring LANE Services

ForeThought 5.0.x allows for LANE 1.0 and 2.0 emulated token rings over ATM. While the ASN-9000 does not offer a token ring media interface or a token ring LEC instance, it can house the LANE services for a token ring emulated LAN on the PowerCell 700.

The token ring service can be selected with the **les add** command when the LES/BUS pair is configured. LES/BUS configuration is discussed in the next section.

```
les add <les-elan-name> <slot> <les/bus-SELbyte> [options]
les add <les-elan-name> <slot> <service-id> [options]
```

The **les add** command has a series of options that can be viewed by entering **help les add** at the atm/lane prompt:

```
18:ASN-9000:atm/lane# help les add
Options:

        -anycast <anycast-atm-address> (anycast address used to contact server)
        -bus <BUS-SELbyte> (BUS selector if it is not the same as LES selector)
        -fwdarp Forward LEARP requests to all clients, even those
        registered as non-proxy.
        -id <ELAN-id> (ELAN identifier in decimal)
        -mtu 1516|1580|4544|9234
        -noregtlvs (set forwarding of registration TLVs to off)
        -peers <peer-atm-address> [<peer-atm-address> ....] (1-10 addresses)
        (ATM address of peer LES in hexadecimal)
        The local LES address must be included in the list of peers.
        In the case of LES created using service-id, local LES address
        will be c5.0005.80ff.e100.0001.<service-id>.002048000001.00.
        -rg <rate-group> (rate group, defaults to 1)
        -ring <ring-number> (token-ring segment identifier in hexadecimal)
        -secure <lecs-atm-address|wka> (secure mode on and LECS address)
        -type ethernet|token-ring
```

To configure for token ring service , enter the **les add** command specifying the ring number and the type as token ring as follows:

3.4.2 Local LES/BUS

A LES/BUS pair configured on the PowerCell 700 is called a local LES/BUS. In FT_5.0.x the LES and BUS are normally configured as a co-located pair. Since a co-located LES and BUS share address resolution information, the forwarding of unknown packets provided by the LES and BUS is optimized. To configure a co-located LES/BUS on the PowerCell 700, use the procedure outlined below.

3.4.2.1 Configuring a LES/BUS Pair

Use the commands described in the following sections to configure a co-located LES/BUS pair. To accommodate LANE v2.0 requirements of MPOA, additional options have been added in FT_5.0. These include the -id option and an increase in MTU. These options can be seen by entering the **help les add** command. In addition to the commands described in the following sections, the ASN-9000 software contains commands for displaying LES/BUS statistics and deleting a LES/BUS configuration.

Configure a LES/BUS pair using the **`les add`** command. The syntax for this command is:

```
les add <les-elan-name> <slot> <les/bus-SELbyte> [options]
les add <les-elan-name> <slot> <service-id> [options]
```

where

<les-elan-name> Specifies a name for this LES. This should be an alphanumeric name from 1 to 40 characters in length.

<slot> Specifies the slot the LES is assigned to. The slot must contain an ATM PowerCell Network Interface Module (NIM).

<les/bus-SELbyte> Specifies the selector byte to be used for the LES and BUS. The selector byte must be specified in hexadecimal and the value must be in the range of 0x80-0xfe.

<service-id> Specifies an 8-digit service identifier. This identifier is used to construct an ATM address where the service can be located independent of the physical topology.

[options] **`-anycast <anycast-atm-address>`**
Specifies an anycast address to be used to contact the server.

`-fwdarp`
Specifies forwarding of LEARP requests to all clients, even those registered as non-proxy.

-id <ELAN-id>
Specifies an ELAN identifier in decimal.

-mtu
Specifies the mtu size. The available mtu sizes are: 1516, 1580, 4544 or9234.

-noregtlvs
If this option is used, forwarding of registration TLVs is set to off.

-peers <peer-atm-address> [<peer-atm-address> ....]<1-10 addresses> Specifies the ATM address of a peer LES in hexadecimal. The local LES address must be included in the list of peers. In the case of a LES created using service-id, local LES address is c5.0005.80ff.e100.0001.<*service-id*>.002048000001.00.

-rg <rate-group>
Specifies the rate group. The rate-group defaults to 1.

-ring <ring-number>
Specifies the ring number in a token-ring segment specified in hexadecimal. The `-ring` option is only valid if the `-type` option is set to token-ring.

-secure <lecs-atm-address|wka>
Specifies that secure mode is set to on and the LECS address.

-type
Specifies the type of LES being configured. The `-type` option is used to start a token ring LEC. The ASN-9000 can provide LANE services for token ring clients but it can't route token ring data. .The available types are ethernet and token-ring.

The following example shows the **`les add`** command:

```
9:ASN-9000: atm /lane# les add engineering 1 0x92
```

3.4.2.1.1 Configuring an Independent NSAP LES

The Independent NSAP LES feature allows the LES's NSAP address to be statically defined rather than dynamically learned via ILMI. Configuring an independent NSAP LES provides redundancy when both a primary and backup connection exists to the network. If the primary link fails, the ASN-9000 will automatically bring the backup PHY on-line and register with the new switch via ILMI but still use static NSAP address. The static SNAP address is the one that should exist in the LECS configuration file.

To configure an independent NSAP LES, issue the following command:

les add ***<les-elan-name> <slot> <Service-ID>***
[[rg=]***<rate-group>*****] [-type](ethernet|token-ring)**
[-mtu(1516|4544|9234)]

<les-elan-name> The ELAN that the LES/BUS pair being created serves. Specify an alphanumeric name from 1 to 32 characters in length.

<slot> The slot that contains the PowerCell 700. Slots are labeled on the chassis. Slot numbers can also be determined by using the **system config show** command.

<Service-ID> Specifies the 8-digit Service Identifier (decimal notation or prefix with a "0x" for hex. notation). This ID is used to construct an ATM address where the service can be located independent of the physical topology. The LES/BUS are created under this usage.

[rg=]<rate-group> Specifies the maximum amount of traffic that can be transmitted over the ATM segments to which the rate group is assigned. The default is 1 group with 155 mbps. Up to 16 rate groups can be defined.

[-type](ethernet|token-ring) Specifies the type of encapsulation in use by the LES.

[-mtu(1516|4544|9234)] Specifies the maximum size of the MTU.

In the following example, a LES for an ELAN called "elan1" is added in slot 2. The service-id is 1111 1111.

```
5: ASN-9000:atm/lane# les add elan1 1 1111 1111.
```

3.4.2.2 Deleting a Configured LES/BUS Pair

To delete a configured LES/BUS, issue the following command:

```
les delete <les-elan-name> <slot>
```

<les-elan-name> The name of the ELAN that the LES serves.

<slot> The slot that contains the PowerCell 700. Slots are labeled on the chassis.

The following example shows the delete command:

```
:
11:ASN-9000:atm/lane# les delete marketing 1
```

3.4.2.3 Displaying the LES/BUS Configuration

After a LES/BUS pair is configured, display the configuration by issuing any of the following command:

```
les [show] <les-elan-name>|all <slot>|<all> [advanced]
```

<les-elan-name>|all The name of the ELAN that the LES/BUS pair serves. If **all** is specified, configuration information about all co-located LES/BUSs in the specified slot is displayed.

<slot>|all The slot that contains the PowerCell 700. Slots are labeled on the chassis. Slot numbers can also be determined by using the **system config show** command. If **all** is specified, configuration information about all co-located LES/BUSs on all modules in the ASN-9000 is displayed.

[advanced] Specifying advanced will produce a display that gives all available information for the ELAN specified or all.

Following are some examples of the information displayed by this command. In the first example, summary information for all the LES/BUS services configured on the ASN-9000 is displayed:

```
65:ASN-9000:atm/lane# les all all
```

```
Slot    Name   Service Type  Service-ID  LES-SEL  BUS-SEL  Rate Group
----    ----   ------------  ----------  -------  -------  ----------
1       tpubs  DLE                -      0x84     0x84      1
1       tester DLE                -      0x88     0x88      1
1       testz  DLE            0x01020304    -        -      1
1       sales  DLE                -      0x95     0x95      1
```

The fields in this display show the following information:

Slot Indicates the ASN-9000 slot that contains the PowerCell 700 on which the listed service is configured.

Name Indicates the name of the ELAN served by the LES/BUS pair.

Service Type Indicates the service type. The service type can be a co-located LES/BUS or DLE. DLE is automatically set up on the ELAN but not implemented until peers are added to the ELAN.

Service-ID Specifies the 8-digit Service Identifier (decimal notation or prefix with a "0x" for hex. notation). This ID is used to construct an ATM address where the service can be located independent of the physical topology. Both the LES and BUS are created under this usage.

LES/BUS-SEL Indicates the Selector byte of the co-located LES/BUS.

Rate Group Specifies the maximum amount of traffic that can be transmitted over the ATM segments to which the rate group is assigned. The default is 1 group with 155 mbps. Up to 16 rate groups can be defined

The command below shows the `les add` command with the advanced option:

```
17:ASN-9000:atm/lane# les TP 1 advanced
ELAN Name: "TP"
  LES:     47.0005.80.ffe100.0000.f21a.1bdd.0000ef039ab1.98
  BUS:     47.0005.80.ffe100.0000.f21a.1bdd.0000ef039ab1.98
LAN Type: Ethernet/IEEE 802.3     Maximum Data Frame Size: 1516
Non-proxy Control Distribute VCC: -.-
Proxy Control Distribute VCC: -.-
Multicast Forward VCC: -.-
Number of local clients: 0
```

The fields in this display show the following information:

LES ATM Address Indicates the ATM address of the LES.

BUS ATM Address Indicates the ATM address of the BUS associated with this LES.

LAN Type Indicates the type of LAN.

Maximum Data Frame Size Indicates maximum data frame size.

Non-proxy Control Distribute VCC Indicates how many ATM hosts served by the LES are not proxies.

Multicast Forward VCC VC used for distributing data from BUS.

Number of Local Clients Indicates number of LECs connected to the LES.

3.4.2.4 Displaying LES/BUS Statistics

To display LES/BUS statistics, issue the following command:

```
stats [show] les  <service-name>|all <slot>|all
```

<service-name>|all .The name of the ELAN that the LES/BUS pair serves. If all is specified, configuration information about all co-located LES/BUS pairs in the specified slot is displayed.

<slot>|all The slot that contains the PowerCell 700. Slots are labeled on the chassis. If **all** is specified, statistics for **all** co-located LES/BUS on all modules are displayed.

Following is an example of the information produced by this command:

```
19:ASN-9000:atm/lane# stats les tpubs 1
LES Statistics for: tpubs
---------------------------------------------------------------------------
LES Statistics:
Join Requests In        :          2
ARP Requests In         :          2
ARP Responses Out       :          2
ARP Requests Forwarded  :          0
Unknown Control In      :          0

BUS Statistics:
Unicast Data In    :          0
Multicast Data In  :        117
Known Control In   :          0
Unknown Control In :          0
```

The fields in this display show the following information:

Join Requests In Indicates a LEC request to join the LES.

ARP Requests In Indicates the number of LE_ARP requests received on this ELAN by the module.

ARP Responses Out Indicates the number of LE_ARP responses sent on this ELAN by the module.

ARP Requests Forwarded Indicates the number of LE_ARP requests forwarded on this ELAN by the module.

Unknown Control In Indicates how many control packets of an unknown type the LES has received.

3.4.2.5 Clearing LES/BUS Statistics

To delete statistics for a configured LES/BUS pair, issue the following command:

```
stats clear les<service-name>|all <slot>|all
```

<service-name>|all Specifies the ELAN or all for which to clear statistics.

<slot>|all Specifies the slot that contains the PowerCell 700. If **all** is specified, statistics for **all** LES/BUS pairs on **all** slots are displayed.

The example shows the command for clearing LES/BUS status:

```
6:ASN-9000:atm/lane# stats clear les tpubs 1
```

3.4.3 Configuring LANE Services

To use the PowerCell module for bridging and routing in a LANE 1.0 or 2.0 ATM network, perform the following configuration tasks:

- Prepare the ATM hardware for LANE. For more information on preparing the ATM hardware, see the *ForeRunner ASN-9000 Installation and Maintenance Manual*.
- Join a LEC to an ELAN by adding the ELAN to the segments allocated to the PowerCell module. The ELAN name assigned to a segment must match the ELAN name specified when configuring the LECS on the ATM switch. (Use the **elan add** command.)

The **elan add** command enables the LEC automatically. Specify the LECS address with either the **elan add** or the **lecs cset** command. By default, the PowerCell module uses LECS Well Known Address (If the LECS is not used to get configuration information, the LES address must be specified when adding an ELAN to the LEC.)

The following procedures are used to configure LANE services.

1. From the atm/lane subsystem, add an ELAN, using the following command syntax:

```
elan add <segment> <elan-name>|-auto [la <les-atm-address> | lu
                         <lecs-atm-address>]
```

where

<segment> Specifies the ATM segment to be configured for LANE services.

<elan-name>|-auto Specifies a name for the elan or -auto. Specifying auto for <elan-name> establishes a Plug-n-Play LANE configuration.

la <les-atm-address>|
lu <lecs-atm-address> Specifies that the following 40-character ATM address is either for the LES (**la**) or LECS (**lu**). The corresponding ATM address entry must be preceded with **la** or **lu** to designate the appropriate segment.

The example below shows the **`elan add`** command:

```
3:ASN-9000:atm/lane# elan add 1.2 marketing
```

2. Configure the PowerCell as a LEC, using the following command:

```
lec cset lecs-addr <lecs-atm-address|wka> <slot> |all
```

NOTE

If the **`lec cset`** command is not used, the slot is assigned the ATM Forum's well-known LECS address (wka) as its default LECS address.

where

<lecs-atm-address|wka> Specifies a valid ATM LECS address or "wka," which is the well-known LECS ATM address.

<slot>|all Specifies the slot to be used by the LEC or all slots containing ATM NIMs.

The example below shows the **`lec cset`** command:

```
5:ASN-9000:atm/lane# lec cset lecs-addr wka 1
```

The display below shows lec configuration for slot 1:

```
4:ASN-9000:atm/lane# lec 1

LEC Configuration For slot: 1
---------------------------------------------------------------------------
LE Client State       : Enabled
LE Client ATM Address : 47.0005.80ff.e100.0000.f21a.1bdd.0000ef039ab1.00
LECS ATM Address      : 47.0079.0000.0000.0000.0000.0000.00a03e000001.00
```

3.4.3.1 Changing ELAN Parameters

Use the **elan set** command to change the default for an ELAN parameter. Display the current settings for each parameter using the **elan show** command.

The LECS configuration file on the FORE ATM switch contains parameters similar to the ELAN parameters maintained by the module. Most of the defaults for the parameters match the defaults for the FORE LECS equivalents to these parameters. FORE Systems recommends that the defaults for these values be used. Setting parameters on the ASN-9000 overwrites values supplied by the LECs. The syntax for the **`elan set`** command is:

```
elan [show] <elan-name>|all
elan set <elan-name>|all arp-aging|aa <time [secs]>
elan set <elan-name>|all bus-rate|br <packets per second>
elan set <elan-name>|all control-timeout|cto <time [secs]>
elan set <elan-name>|all flush-timeout|fto <time [secs]>
elan set <elan-name>|all forward-delay|fd <time [secs]>
elan set <elan-name>|all max-arp-retry|mar <count>
elan set <elan-name>|all vcc-timeout|vto <time [secs]>
```

<elan-name>|all Specifies the name of the ELAN for which to set ELAN parameters. If **`all`** is specified, the parameters are applied to all ELANs in the ASN-9000.

Specify one of the following parameters with the **`elan set`** command and the value to assign to the parameter:

arp-aging|aa <time> Specify from **`10`** through **`300`** seconds. The default is **`300`**. The corresponding parameter in the LECS file is `.Aging_Time`.

bus-rate|br <packets per second> Specify from **`0`** through **`10`** Packets Per Second. The default is **`1`**. The corresponding parameter in the LECS file is `.Maximum_Unknown_Frame_Time`.

<control-timeout|cto <time> Specify from **`10`** through **`600`** seconds. The default is **`120`**. The corresponding parameter in the LECS file is `.Control_TimeOut`.

flush-timeout|fto <time> Specify from **`1`** through **`10`** seconds. The default is **`6`**. The corresponding parameter in the LECS file is `.Flush_TimeOut`.

<forward-delay|fd <time> Specify from **`4`** through **`30`** seconds. The default is **`15`**. The corresponding parameter in the LECS file is `.Forward_Delay_Time`.

max-arp-retry|mar <count> Specify from **0** through **2** requests. The default is **2**. The corresponding parameter in the LECS file is `.Maximum_Retry_Count`.

vcc-timeout|vto <time> Specify from **1** through **720** minutes. The default is **20**. The corresponding parameter in the LECS file is `.VCC_TimeOut_Period`.

3.4.3.2 Displaying a LE_ARP Table for an ELAN

The ASN-9000 maintains a separate LE_ARP table for each ELAN on the module. The LE_ARP table maps the MAC addresses of the devices in an ELAN to their corresponding ATM addresses. To display the LE_ARP table for an ELAN, issue the following command:

at [show] elan=*<elan-name>*|addr=*<mac-address>*|all

elan=<elan-name>|addr=<mac-address>|all Specifies an ELAN name or MAC address. If **all** is specified, LE_ARP table entries for all ELANs are displayed.

Following are some examples of the information displayed by this command. In the following example, the LE_ARP table entries for “elan1” are displayed.

```
5:ASN-9000:atm/lane# at show elan=elan1

ARP Table For ELAN: elan1
Seg. MAC Address         ATM Address
---------------------------------------------------------------------------
 1.3  00-20-48-1a-1e-3c   47:0005:80ff:e100:0000:f21a:1e3c:0020481a1e3c:10
 1.4  00-20-48-1b-1f-3d   47:0005:80ff:e100:0000:f21a:1e3c:0020481a1e3c:02
```

In the following example, a MAC address is specified. The ELAN name and ATM address corresponding to the MAC address are listed.

```
6:ASN-9000:atm/lane# at show 00-20-48-1a-1e-3c
ARP Table For MAC Address: 00-20-48-1a-1e-3c
Seg. Elan Name             ATM Address
 ---------------------------------------------------------------------------
 1.9  elan1                 47:0005:80ff:e100:0000:f21a:1e3c:0020481a1e3c:10
```

The fields in this display show the following information:

Seg Indicates the ASN-9000 segment on which the ELAN is configured.

MAC Address Indicates the MAC address of a LANE device.

Elan Name Indicates the name of the ELAN.

ATM Address Indicates the ATM address of a LANE device.

3.4.3.2.1 Clearing the LE_ARP Table

To clear an ELAN’s LE_ARP table, issue the following command:

```
at clear <elan-name>|all
```

<elan-name>|all Specifies the name of the ELAN to clear the LE_ARP table. If **all** is specified, LE_ARP tables for all ELANs are cleared.

3.4.3.3 Displaying the Virtual Circuits on an ELAN

To display the VCs (Virtual Circuits) in use on an ELAN, issue the following command:

```
vt [show] <elan-name>|all
```

elan-name>|all Specifies the name of the ELAN to display the VC table. If **all** is specified, active VCs on all ELANs are displayed.

Following is an example of the information displayed by this command. In this example, the VC for “elan1” is displayed.

```
7:ASN-9000:atm/lane# vt elan1
VC Table For ELAN: elan1

 Seg. MAC Address         VC
-----------------------------------------------------------------
  19  00-20-48-1a-1e-3c    67
```

The fields in this display show the following information:

Seg Indicates the ASN-9000 segment on which the ELAN is configured.

MAC Address Indicates the MAC address of the device at the other end of the VC.

VC Indicates the number of the VC connecting the ELAN on the module to the other device. The VC number is negotiated by the module and the ATM switch when the VC is established. The VC number is different for each VC.

3.4.3.4 Displaying Statistics

The ASN-9000 collects statistics for the data traffic and the control traffic sent and received by the ELANs configured on the module. All statistics are collected on a per-ELAN basis.

- Statistics for data traffic are collected as interface statistics because the ELAN data packets are processed on the interface layer.
- Statistics for ATM control traffic are collected as ELAN statistics because they are processed on the ELAN layer.

To display ATM statistics, issue the following command:

`stats [show] elan` *`<elan-name>`***`|all elan|if|all`**

<elan-name>|all Specifies the name of the ELAN to display statistics. If **`all`** is specified, statistics for all the ELANs in the ASN-9000 are displayed.

elan|if|all Specifies the statistics you want displayed.

`elan` Displays ELAN (control) statistics.

`if` Displays interface (data) statistics.

`all` Displays ELAN and interface statistics.

Here is an example of the elan statistics displayed by this command. In this example and the following example, the statistics are displayed for an ELAN named “tpubs.”

```
28:ASN-9000:atm/lane# stats elan tpubs elan

ELAN Statistics For ELAN: tpubs
---------------------------------------------------------------------------
SVC Failures      :          0
Total Control In  :          3
Total Control Out :          3
ARP Replies In    :          0
ARP Replies Out   :          0
ARP Request In    :          0
ARP Request Out   :          0
Join ELAN Calls   :          1
```

The fields in this display show the following information:

SVCs Failures Indicates the number of Switched Virtual Circuits (SVCs) that have been released (torn down) by this ELAN. An SVC is released after the two ends of the SVC (the ELAN on the module and the device on the other end) stop exchanging traffic.

Total Control In Indicates the number of LANE 1.0 or 2.0 control packets received on this ELAN by the module.

Total Control Out Indicates the number of LANE 1.0 or 2.0 control packets sent on this ELAN by the module.

ARP Replies In Indicates the number of LE_ARP replies received on this ELAN by the module.

ARP Replies Out Indicates the number of LE_ARP replies sent on this ELAN by the module.

ARP Request In Indicates the number of LE_ARP requests received on this ELAN by the module.

Arp Request Out Indicates the number of LE_ARP requests sent on this ELAN by the module.

Join ELAN Calls Indicates the number of times an ELAN on the module has requested to join the like-named ELAN configured on the ATM switch.

The following stats command produces interface statistics:

```
18:ASN-9000:atm/lane# stats elan tpubs if

Interface Statistics For ELAN: tpubs

---------------------------------------------------------------------
Out MCast Pkts    : 0
Out Errors        : 0
Out Discard       : 0
Out UCast Pkts    : 0
In MCast Pkts     : 1
In Errors         : 0
In Discard        : 0
In UCast Pkts     : 0
In Unknown Protos : 0
MTU size          : 1516
```

The fields in this display show the following information:

Out MCast Pkts Indicates the number of Ethernet broadcast or multicast packets the ELAN has sent to the ATM network.

Out Errors Indicates the number of Ethernet packets sent by the ELAN that experienced an error during transmission.

Out Discard Indicates the number of Ethernet packets that were discarded due to an error on the module, rather than sent to the ATM network.

Out UCast Pkts Indicates the number of Ethernet unicast packets the ELAN has sent to the ATM network.

In MCast Pkts Indicates the number of Ethernet broadcast or multicast packets the ELAN has received from the ATM network.

In Errors Indicates the number of Ethernet packets received by the ELAN that contain errors.

In Discards Indicates the number of Ethernet packets that were discarded due to an error in the module, rather than received for the ELAN.

In UCast Pkts Indicates the number of Ethernet unicast packets the ELAN has received from the ATM network.

In Unknown Protos Indicates the number of Ethernet packets received that were using an unknown protocol.

MTU size Indicates the MTU (maximum transmission unit) for the protocol being used in the ELAN. For Ethernet, the MTU is 1514 bytes

3.4.3.5 Clearing Statistics

To clear ATM statistics, issue the following command:

```
stats clear elan <elan-name>|all
```

<elan-name>|all Specifies the name of the ELAN for which to clear statistics. If **all** is specified, statistics for all the ELANs in the ASN-9000 are cleared.

3.4.3.6 Verifying the LANE Configuration

After configuring the module as a LEC, the LEC software on the module serves all the ATM segments on that module configured for LANE 1.0 or 2.0. To show the LEC that is to be used by all the LECs, issue the following command:

```
lec [show] <slot>|all
```

<slot>|all Specifies the slot that contains the module.

Slots are labeled on the chassis. Slot numbers can also be determined by using the **system config show** command. If **all** is specified, the LEC configuration for all the ASN-9000 modules in the chassis is displayed.

Following is an example of the information displayed by this command. The “LE Client” is the LEC (module).

```
20:ASN-9000:atm/lane# lec show 1

LEC Configuration For slot: 1
---------------------------------------------------------------------------
LE Client State       : Enabled
LE Client ATM Address : 47.0005.80ff.e100.0000.f21a.1bdd.0000ef039ab1.00
LECS ATM Address      : 47.0079.0000.0000.0000.0000.0000.00a03e000001.00
```

The fields in this display show the following information:

LE Client State Indicates whether the module is enabled as a LEC.

LE Client ATM Address Indicates the ATM address of the module. The address displayed is the base address of the LEC. The Selector byte contains zeroes (00). Each ELAN configured on the module uses the base address but has a unique value in the Selector byte.

LECS ATM Address Indicates the ATM address of the LECS (LAN Emulation Configuration Server).

3.4.3.7 LANE 1.0 and 2.0 Configuration Examples

Depending upon the ATM hardware and software, these items can be configured on the ATM switch, on a workstation, or the configurations might be distributed between the ATM switch and a workstation. For example, the LECS may be installed on a Sun workstation and the LES/BUS on the FORE switch, as shown in Figure 3.11.

NOTE In software version 7-2.6.4.0 and later, configure the LES and BUS on the PowerCell 700. Figure 3.11 shows the LES and BUS configured on the ATM switch.

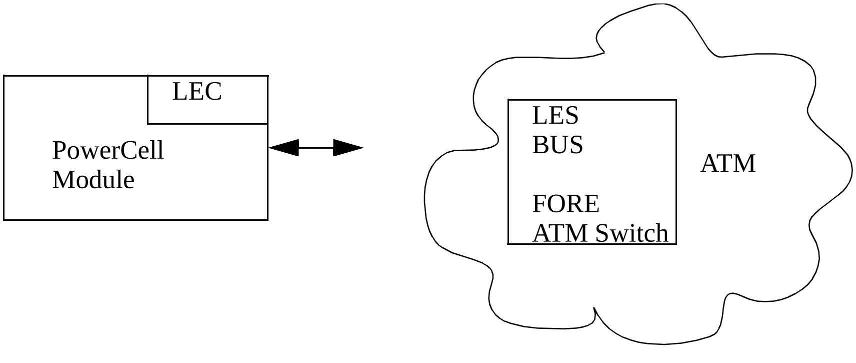


Figure 3.11 - Example of LECS Configuration on a Sun Workstation

Table 3.1 lists the hardware used for each LANE component.

Table 3.1 - Lane Component Hardware Table

LANE Component	Hardware	Software
LEC	PowerCell module	ELAN names. Each ELAN is associated with an ATM segment. If no LECS is configured, the ATM address of the LES is specified with each ELAN name.
LES/BUS	ASX-200BX ATM Switch	ATM address of the LES and BUS.
LECS	SBA-200 ATM Adapter Card (installed on Sun workstation)	LECS configuration file. Contains the ATM addresses of the ELANs. Also contains filters for accepting or discarding specific ATM addresses.

NOTE

If a LECS is unavailable or you cannot add a LECS, you can still use ELANs if you supply the LES address when you add the ELANs.

3.4.3.8 LEC Example

Before segments can begin switching in the LANE environment, add an ELAN to each segment as shown in the following example. The terse form of the **`elan add`** command is used.

```
13:ASN-9000:atm/lane# elan add 1.9 elan1|0
14:ASN-9000:atm/lane# elan add 1.9 elan1|1
15:ASN-9000:atm/lane# elan add 1.9 elan1|1
```

When adding the ELANs to the segment using the **`elan add`** command, the ELAN is automatically enabled to get information from the LECS, and the LEC software is started on the module.

3.4.4 Distributed LAN Emulation (DLE)

Distributed LAN Emulation (DLE) allows you to configure ELANs for redundancy in a network configuration. As such it replaces the LECS Failover mechanism that existed in FT_4.0.0. DLE implementation makes all clients on an ELAN appear to be connected to a single LANE LES/BUS pair. If a LES fails on one device in a network configuration but the LES on the PowerCell of that device is configured for DLE, a LES on another device in the network will take over and clients connected to the failed device will establish connections to that backup LES.

DLE offers several significant advantages over using a single server for an ELAN:

- **Scalability**: DLE peer servers distribute the circuit and processsing load. The number of LANE LAN emulation clients is no longer limited by the number of circuits one LES/BUS platform can maintain, since many platforms can support a single ELAN.
- **Distributed workgroups**: ELANs with groups of LAN emulation clients in different locations can be designed for higher performance by providing a DLE peer server with each group. Having a closer server allows broadcasts and address resolution within each group to improve.
- **Reliability**: In a single ELAN, the server is a single point of failure. If the server fails, clients in the ELAN are unable to discover each other through broadcast queries and are unable to resolve MAC addresses into ATM addresses. Increased network reliability, therefore, requires that ELANs have backups for LES and BUS functions.

3.4.5 Configuring DLE

To configure DLE on the ASN-9000, use the **`les add`** command, specifying the anycast address and the DLE peer addresses:

```
les add <les-elan-name> <slot> <sel byte> -anycast <DLE anycast
  address> -peers <DLE peer address><DLE peer address> . . .
```

`where`

<les-elan-name> Specifies the name for ELAN to which this LES belongs. This should be an alphanumeric name from 1 to 40 characters in length.

<slot> Specifies the slot the LES is assigned to. The slot must contain an ATM PowerCell Network Interface Module (NIM).

<sel-byte> Specifies the selector byte to be used for the LES and BUS. The selector byte must be specified in hexadecimal and the value must be in the range of 0x80-0xfe.

<-anycast> Specifies the LES Anycast ATM Address used by this ELAN.

<-peers> Specifies the ATM address of a peer LES in hexadecimal. The local LES address must be included in the list of peers. In the case of a LES created using service-id, local LES address is c5.0005.80ff.e100.0001.<*service-id*>.002048000001.00. There can be a maximum of 10 peers.

```
11:ASN-9000:atm/lane# les add marketing 1 0x81
-anycast C5.0005.80.ffe100.0000.f21a.21b8.0097036324b2.25
-peers 47. 0005. 80. ffe100.0000.f21a.10bb.0000ef062990.82
47. 0005. 80. ffe100.0000.f21a.3552.0000ef068329.81
47. 0005. 80. ffe100.0000.f21a.3218.0000ef083632.44
```

3.4.6 Distributed LAN Emulation Model

To address the limitations of the single server model, DLE distributes the LANE services load among a mesh of LES/BUS DLE peer servers, as shown in Figure 3.12.

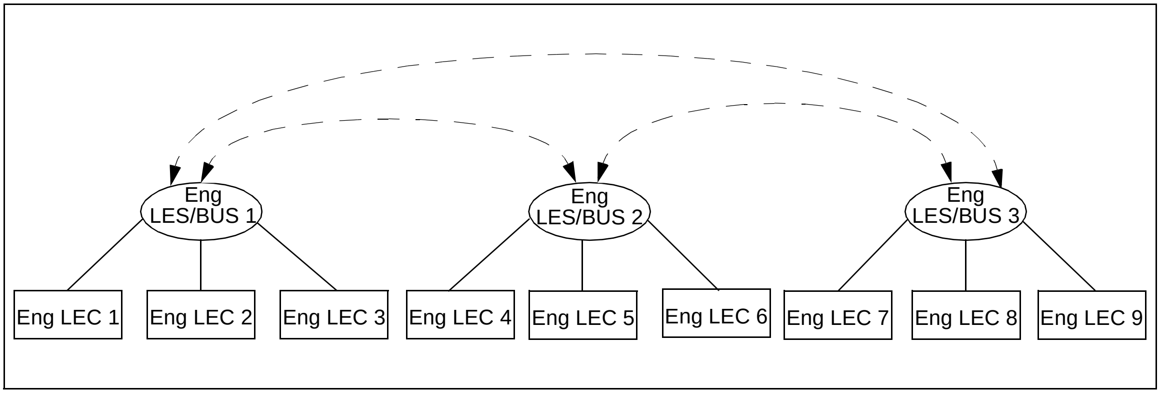


Figure 3.12 - Distributed LAN Emulation Model

Each DLE peer server actually maintains two sets of connections: one is a point-to-multipoint connection to each of its peers for broadcasting multicast data and flooding control information, and the other includes individual point-to-point connections to each peer for directed control traffic.

Each DLE peer server that supports the ELAN is responsible for registering and giving reports about the LECs that are attached to it directly. Each DLE peer server propagates this information to both its locally attached LECs and its peers.

NOTE

Each device running a DLE peer server must use *ForeThought* 5.0 or greater; however, the DLE peer servers support clients and attached switches using *ForeThought* 4.0 and 4.1, and third-party devices that are ATM Forum LANE 1.0 compliant.

3.4.6.1 Using DLE

Figure 3.13 shows how a connection begins to be established through DLE peer servers. LEC 1 wants to communicate with LEC 9, which is in the same ELAN, but is locally attached to a different DLE peer server. First, ❶ LEC 1 sends an IP ARP broadcast request to its local DLE BUS. Then, ❷ the BUS broadcasts the packet to both its locally attached LECs and its DLE peer servers.

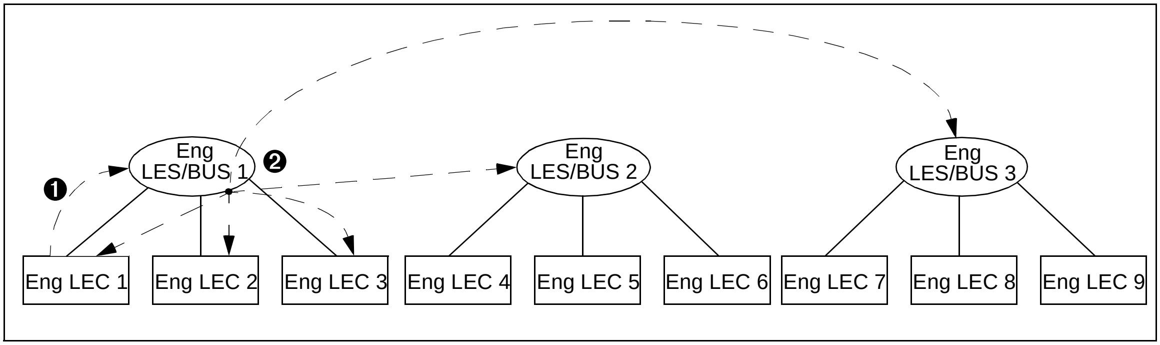


Figure 3.13 - IP ARP Broadcast from LEC 1 to LEC 9

Upon receiving the broadcast from the first DLE peer server, the peers re-distribute the packet to their own locally attached LECs ❸, as shown in Figure 3.14, so the packet arrives its actual destination at LEC 9.

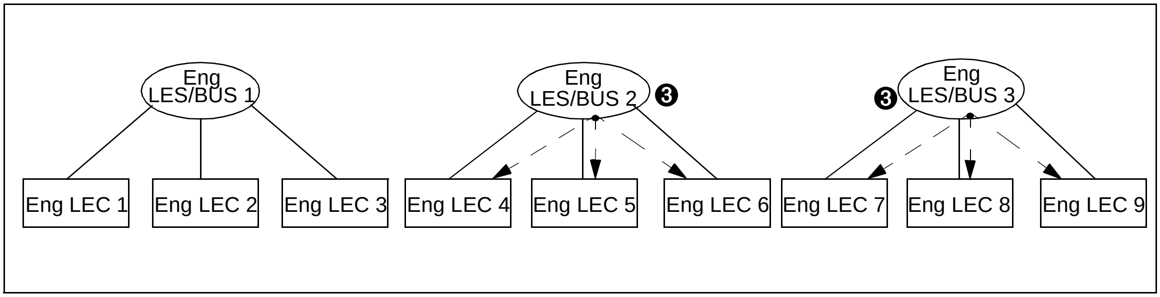


Figure 3.14 - Re-distributing the Broadcast across DLE Peer Servers

NOTE

The peers do not re-distribute the packet to other peers; this would create a loop.

LEC 9 recognizes its IP address, and prepares an IP ARP response. As shown in Figure 3.15, it then sends an LE-ARP request to its local LES ❹, asking for the ATM address that matches LEC 1’s MAC address. Since LEC 9’s local LES does not have an entry for LEC 1, the local LES passes the query along to all of its locally-attached proxy LECs (none are shown in this figure) and all of its DLE peer servers ❺.

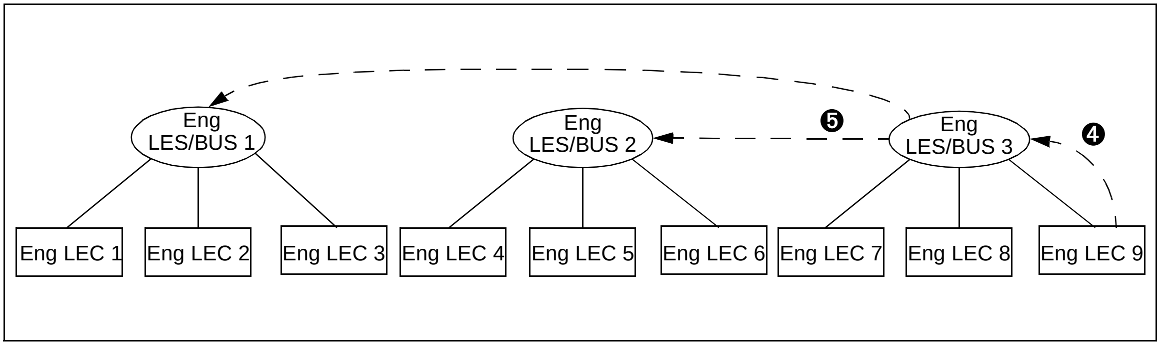


Figure 3.15 - LE-ARP for Unknown Host Sent to Proxies (not shown) and DLE Peer Servers

In Figure 3.16, the second DLE peer server is attached to two proxy LECs (LEC 4 and LEC 5). When the DLE peer server receives the LE-ARP query, it cannot resolve the query, so the DLE peer server re-distributes the query to its proxy LECs ❻ (but not to its peer servers again, to avoid a loop). Meanwhile, the first peer server has been able to resolve the LE-ARP for the address of LEC 1 and has sent an LE-ARP response to the third server ❼.

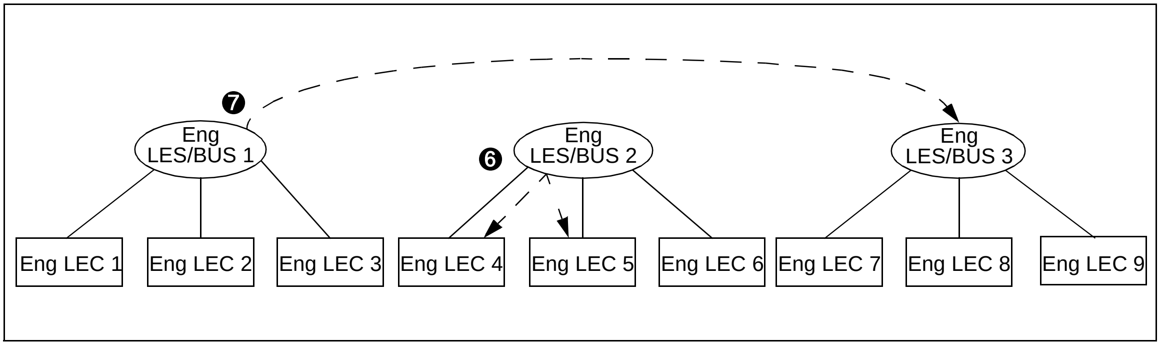


Figure 3.16 - LE-ARP Query Answered by One DLE Peer Server and Re-distributed by Another

When the third DLE peer server receives the LE-ARP response, it passes it directly to LEC 9 (which sent the original query) ❽. The third DLE peer server also caches the registration information for LEC 1 so that other local LECs do not have to go through the entire process again. However, this cache ages out over time. LEC 9 can now open a connection to LEC 1, and send its IP ARP response ❾, as shown in Figure 3.17.

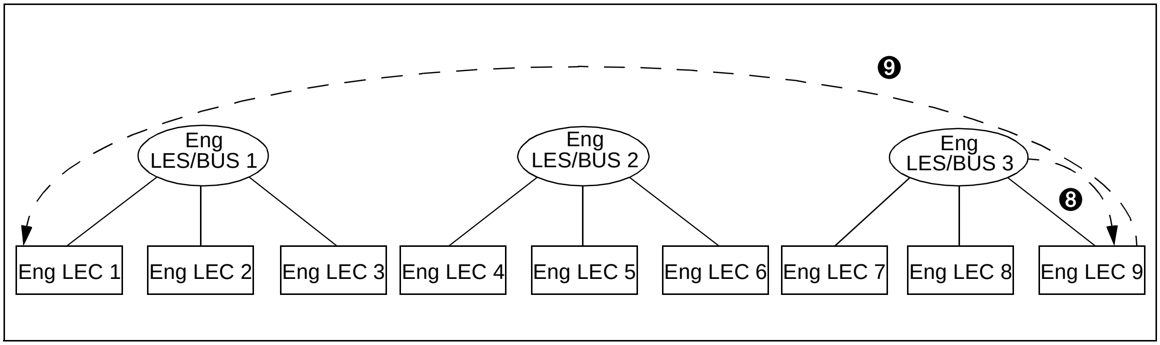


Figure 3.17 - LE-ARP Response Delivered and LEC 9 Contacts LEC 1

3.5 ELAN Access Control

Basic ATM Forum LAN Emulation Servers do not guard against unauthorized users learning an ELAN’s LES address and then joining the ELAN. However, a method of authorization checking is available. After a LEC obtains the address of its LES, the LEC sends a request to the LES to join the ELAN. If the LES has ELAN access control enabled, it sends a message to the LECS to verify that the LEC is allowed to join. If verification is received from the LECS, then the LES gives the LEC permission to join. If verification is not received from the LECS, the LES rejects the join request and the LEC is dropped.

Using this feature, an authorization check is also performed each time the LECS reloads the LECS configuration file. (The LECS periodically checks whether its configuration file has been modified, and, if it has, the file is re-read. The length of this period, in seconds, is defined by the `Reload_Period` key.) If the file has changed to disallow some clients that were previously allowed, those clients will be dropped from the ELAN.

NOTE

ELAN access control also works with a third-party LECS. The LES revalidates the client every 600 seconds since the third-party LECS will not contact the LES with configuration changes.

You can enable ELAN access control when you are creating the LES. When you use the **`les add`** command, specify the **`-secure`** option. This indicates you want to activate a secure LES/BUS pair.

```
les add <les-elan-name> <slot> <les/bus-SELbyte> [options]
les add <les-elan-name> <slot> <service-id> [options]
```

```
Options:
        -anycast <anycast-atm-address> (anycast address used to contact server)
        -bus <BUS-SELbyte> (BUS selector if it is not the same as LES selector)
        -fwdarp Forward LEARP requests to all clients, even those registered as nonproxy.
        -id <ELAN-id> (ELAN identifier in decimal)
        -mtu 1516|1580|4544|9234
        -noregtlvs (set forwarding of registration TLVs to off)
        -peers <peer-atm-address> [<peer-atm-address> ....] (1-10 addresses)
        (ATM address of peer LES in hexadecimal)
        The local LES address must be included in the list of peers.
        In the case of LES created using service-id, local LES address
        will be c5.0005.80ff.e100.0001.<service-id>.002048000001.00.
        -rg <rate-group> (rate group, defaults to 1)
        -ring <ring-number> (token-ring segment identifier in hexadecimal)
        -secure <lecs-atm-address|wka> (secure mode on and LECS address)
        -type ethernet|token-ring
```

If you enter **wka** with the **-secure** option, the ATM Forum well-known LECS address is used. In this case, you do not have to type the actual well-known address. However, if you are using an LECS address that is different than the well-known address, then you must type the full LECS ATM address to be used.

3.6 Configuring LANE/MPOA

3.6.1 Overview

The main function of Multi-Protocol over ATM (MPOA) is to provide the best routes for connectivity over ATM networks. To accomplish speed and efficiency of data transfer, MPOA utilizes the strengths of ATM network topology and configuration, such as virtual circuits support, to effectively link up shortcuts between a source and destination. A shortcut is a direct one-hop path to a destination or to the nearest transit point to a destination.

This section describes the commands for configuring MPOA on the ASN-9000 platform. The MPOA commands are grouped functionally into two categories: Multi-Protocol Server (MPS) and Next Hop Server (NHS). The MPS commands function over LAN Emulation (LANE), whereas NHS commands function over IP-Over Non-Broadcast, Multi-Access (ION-NBMA).

3.6.1.1 MPOA Shortcuts

This section explains how shortcuts are established in a network. The actual commands and directions for configuring MPSs, MPCs, and NHSs are given in separate sections following this general description.

For a shortcut to be established, an ingress MPC must first have been configured on the originating device, all switches connecting the originating device to the terminating device must have been configured with MPSs, and the terminating device must a have a MPC or MPS configured. See Figure 3.18 below.

The MPC on the terminating hub must be on a segment that is configured as a bridged connection to the port behind which the destination resides. For an MPC to initiate or terminate a shortcut, it must be bridged to the segment on which the traffic originates. A non-MPOA enabled ATM segment cannot create shortcuts. Shortcuts are triggered by a MPC only for its bridged legacy ports. Both the inbound and outbound LANE segments must have a MPS configured. MPSs must be configured on the LANE path on all routed hops to the destination.

When a packet enters at the ingress MPC, it will be bridged via LANE to the next hop MPS, which connects with the next hop MPS/MPC over a LANE segment. Each packet sent to a specific IP address is counted, and when a certain threshold is reached the MPC is required to send a request for the ATM address of the MPC/MPS closest to the destination address to be used for establishing a shortcut to a specific downstream MPS. MPOA shortcuts can terminate at MPCs or MPSs, but an MPC is required to trigger a shortcut.

If a shortcut has been established, the ingress MPC strips the DLL encapsulation from the packet and sends it via the shortcut. When the packet arrives via shortcut at the egress MPC, it is examined and either a matching egress cache entry is found or the packet is dropped. All encapsulated information is stored at the egress MPC/MPS and is inserted at the egress point before being passed on to legacy ports.

In Figure 3.18 below, the ingress MPC sends a packet via ELAN A through the ASX 1000 switch to the MPS on the ASN-9000. The ASN-9000 routes the packet to the MPS on ELAN B, which forwards the packet via ELAN B to the egress MPC on the terminating switch. Once the shortcut has been established, the MPC caches the information in the ingress MPC Cache, sets up a shortcut VCC, and forwards packets for the destination over the shortcut.

.

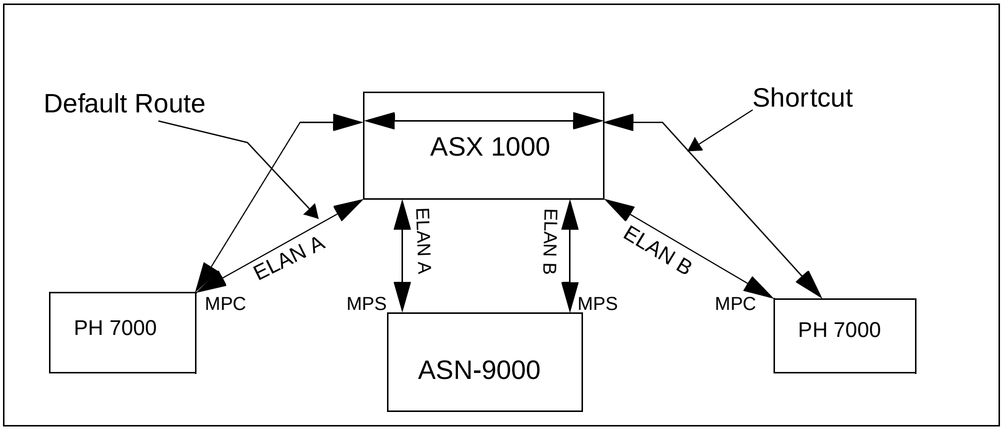


Figure 3.18 - MPOA shortcut

3.6.1.2 Multi-Protocol Server (MPS)

An MPS is the logical component of a switch that provides internetwork layer forwarding information to MPCs. A full NHS, as defined in the Next Hop Resolution Protocol (NHRP), is included in the MPS. The MPS interacts with the local NHS to answer MPOA queries from Ingress MPCs and provide encapsulation information to Egress MPCs.

An MPS converts between MPOA requests and replies and NHRP requests and replies on behalf of the MPCs.

3.6.1.3 Examples of MPOA-Enabled Devices

The following list contains examples of what are considered to be MPOA-enabled devices:

- MPOA Edge Device (including the MPC, LEC, and a bridge port)
- MPOA Host (including the MPC, LEC, and an internal host stack)
- ASN-9000 (including the MPS, which in turn includes a NHS, LEC, and the routing function)

3.6.2 MPS Commands

The following paragraphs describe the commands that have been added to support the MPOA release of ASN-9000software. These commands are used to setup and control configured MPS interfaces.

3.6.3 Interface

The **`interface`** command is used to configure a MPS on a segment. The MPS requires that the PowerCell segment be configured for LANE; otherwise, MPS creation fails with an error. The command syntax for configuring a MPS on a segment is as follows:

```
it|interface add <segment> <sel-byte>
it|interface enl|enable <seglist>|all
it|interface dis|disable <seglist>|all
it|interface del|delete <segment>
```

`where`

add Adds a MPS interface on the specified segment.

sel-byte A selector byte above 0x80-0xfe must be specified for the MPS to control the ATM address. During creation, the selector byte is saved and used during the enable.

Before adding an MPS on a segment the following checks are made:

- The specified segment must be an ATM segment.
- An MPS or NHS should not exist on this segment.
- A LEC must first be configured on this segment.

enl|enable dis|disable Enables/disables the MPS interface(s) on the specified segment-list or all MPS segments. A LEC must be operational on this segment for the MPS to be operational.

del|delete Deleted the MPS interface(s) on the specified segment(s) or all.

The example below shows the **`it add`** command used to configure a MPS on a segment:

```
56:ASN-9000:atm/mps# it add 1.8 0x92
```

Entering **`interface`** displays the following:

```
97:ASN-9000:atm/mps# interface
Segment State       Mode   Control ATM Address
---------------------------------------------------------------------------
1.8 Initial     LECS   (null)
98:ASN-9000:atm/mps#
```

3.6.4 MPS Set

The **mset|mpsset** command sets configuration options for the MPS on the specified segment. The syntax for this command is:

```
mset|mpsset <keyword> <value> <segment>
```

where

<keyword> <value>

selbyte|sel Selector byte value in hexadecimal for MPS control ATM address. This value must be in the range 0x80-0xfe.

keepalivetime | kt Keep alive time entered in seconds. The range is 1 to 300secs. The default value is 10secs. Keepalive time must be less than 1/3 of keepalive lifetime.

keepalivelifetime|klt Keepalive lifetime (range 2-1000 secs).

[config]mode Configuration mode to be used during the next restart of the MPS. Specify either auto[matic] or man[ual]. Automatic causes the MPS to use the LECS for configuration parameters.

cachesize|cs Specifies a maximum number of cache entries (default is 8000).

ilimit Sets the imposition table limit (default is 2000).

<segment> Specifies the segment the MPS resides on to apply the above values.

3.6.5 Configuration Display

The **config [show]** command displays configuration parameters currently in use by the configured MPS interfaces or those in the specified <seglist>. These values are taken from the mps Status Table in the FORE-specific MPOA MIB.

```
config [show] [-m] <seglist>|all
```

where

-m Displays the manual configuration currently set to be used when the config mode is manual (or) to override the LECS values.

<seglist>|all Specifies which segment(s), or all segments for which to display configuration information.

Entering **`config 1.2`** displays the MPS configuration on segment 1.2.

```
59:ASN-9000:atm/mps# config 1.2

MPOA Server Configuration on Segment 1.2
---------------------------------------------------------------------------
Current State            : Initial
Configuration Mode       : LECS
Control Address          : (null)
Authentication Type      : None
Keep Alive Time (sec)    : 10
Keep Alive Lifetime (sec): 35

60:ASN-9000:atm/mps#
```

Entering **`config -m 1.2`** displays the manual configuration on segment 1.2.

```
60:ASN-9000:atm/mps# config -m 1.2
MPS Manual Configuration on Segment 1.2
---------------------------------------------------------------------------
Configuration Mode       : LECS
Control Address          : 00.0000.0000.0000.0000.0000.0000.000000000000.82
Authentication Type      : None
Keep Alive Time (sec)    : 10
Keep Alive Lifetime (sec): 35

61:ASN-9000:atm/mps#
```

where

Current State Displays the current state of the MPS configured on the specified segment, segments in a segment list or all configured MPS segments.

Configuration Mode Displays the configuration mode, which can be one of LES, LES/BUS or LECS.

Control Address Displays the assigned control address if the MPS was manually configured. Otherwise, null is displayed.

Authentication Type Displays the authentication type assigned to the MPS through the `mset|mpsset` command.

Keep Alive Time (sec) Displays the keep alive time, in seconds, as specified in the `mset|mpsset` command.

Keep Alive Lifetime (sec) Displays the keep alive lifetime, in seconds, as specified in the `mset|mpsset` command.

3.6.6 Configuring an MPS

The steps in the following procedure configures an MPS on the ASN-9000.

1. Before configuring an MPS, set up an elan on the segment by entering the **`elan add`** command from the atm/lane subsystem:

```
52:ASN-9000:atm/lane# elan add 1.5 elan5
```

2. Configure an MPS on a segment using the **`it add`** command from the `atm/mps` subsystem.

```
it|interface add <segment> <sel-byte>
```

```
35:ASN-9000:atm/mps# it add 1.5 0x85
MPS Added on Segment 1.5
36:ASN-9000:atm/mps#
```

3. Enable the MPS using the **`it enable`** command from the `atm/mps` subsystem.

```
36:ASN-9000:atm/mps# it enl 1.5
37:ASN-9000:atm/mps#
```

The above steps have configured an MPS on segment 1.1 attached to a vlan called test1. This can be verified by displaying the MPS configuration.

```
3:ASN-9000:atm/mps# config

MPOA Server Configuration on Segment 1.5
---------------------------------------------------------------------------
Current State            : Operational
Configuration Mode       : LECS
Control Address          : (null)
Authentication Type      : None
Keep Alive Time (sec)    : 10
Keep Alive Lifetime (sec): 35
4:ASN-9000:atm/mps#
```

3.6.7 MPS Imposition Table

The **`itable`** command is used to display information on the impositions made by this MPS on the specified segment(s) or all segments. This information comes from the mps Imposition Table specified in the FORE-specific MPOA MIB. The syntax for this command is:

```
itable [show] <seglist>|all
```

Entering **`itable 1.2`** displays the MPS Imposition entries for segment 1.2:

```
59:ASN-9000:atm/mps# itable 1.5
CacheID HoldTime State   Destination
------- -------- ------- -------------------------------------------
123     10       Purged  47.005.80FFE1000000F21A00FB.0020481A02FA.01
134     23       Imposed 47.005.80FFE1000000F21A00FB.0020481A02FA.02
135     10       Pending 47.005.80FFE1000000F21A00FB.0020481A02FA.03
60:ASN-9000:atm/mps#
```

3.6.8 Trace Level

The **`tracelevel|trl`** command is used to set a trace level to a specified level-name on a specified segment, segment list, or all MPS configured segments. The syntax for this command is:

```
set tracelevel|trl level-name <seglist>|all
```

where

level-name Specifies a level-name of info, notice, or warning.

<seglist>|all Specifies the segment, list of segments, or all segments to which to apply the specified level-name.

The command shown in the example below sets the trace level to information on segment 1.5:

```
97:ASN-9000:atm/mps# set trl info 1.5
```

3.6.9 Trace Class

The **traceclass|trc** command is used to enable or disable class-names on a specified segment, segment list or all MPS configured segments. The syntax for this command is:

```
enable traceclass|trc class-name <seglist>|all
disable traceclass|trc class-name <seglist>|all
```

where

class-name Specifies a class-name of rxpkt, txpkt, route, timer or gen.

<seglist>all Specifies the segment, list of segments, or all segments to which to apply the specified class-name.

Enter the command as shown in the example below:

```
95:ASN-9000:atm/mps# enable trc route 1.5
```

3.6.10 Trace Settings

The **tracesettings|tr** command displays tracing on specified segment(s). The syntax for this command is as follows:

```
[show] tracesettings|tr <seglist>|all
```

where

<seglist>all Specifies which segment, list of segments, or all segments to display trace settings.

Entering **tr 1.2** displays the trace activity on segment 1.5.

```
94:PASN-9000:atm/mps# tr 1.5
Entity         Action   Level        Enabled classes
---------------------------------------------------------------------------
MPS :
               Print    (warning)    gen route txpkt rxpkt
               Log      (debug)      gen route txpkt rxpkt
mps 1.5 :
               Print    (notice)     route
               Log      (notice)     route
```

3.6.11 Statistics

The **stats** command is used to display the MPS interface statistics of a MPS configured on a specified segment, a list of segments, or all MPS configured segments. The syntax of this command is as follows:

```
stats [show] [-e] <seglist>|all
   stats clear <seglist>|all
```

where

[-e] Displays only error statistics on the specified segment, list of segments, or all MPS configured segments.

clear Clears statistics on the specified segment, list of segments or all MPS configured segments.

<seglist>all Specifies the segment, list of segments, or all segments for which to display trace settings.

Entering **stats 1.2** from the atm/mps subsystem displays the following type of information:

```
78:ASN-9000:atm/mps# stats 1.2
MPOA Server Statistics on Segment 1.2
---------------------------------------------------------------------------
MPOA Resolution Requests Received           : 0
MPOA Resolution Reply Acks Transmitted      : 0
MPOA Resolution Reply Naks Transmitted      : 0
MPOA Cache Imposition Replies Received      : 0
MPOA Cache Imposition Requests Transmitted  : 0
MPOA Egress Cache Purge Requests Received   : 0
MPOA Egress Cache Purge Replies Transmitted : 0
MPOA Keep Alives Transmitted                : 0
MPOA Triggers Transmitted                   : 0
NHRP Resolution Requests Received           : 0
NHRP Resolution Requests ReInitiated        : 0
NHRP Resolution Requests Forwarded          : 0
NHRP Resolution Replies Received            : 0
NHRP Resolution Reply Naks Transmitted      : 0
NHRP Resolution Reply Acks Transmitted      : 0
NHRP Resolution Replies Forwarded           : 0
NHRP Purge Requests Received                : 0
NHRP Purge Requests Transmitted             : 0
NHRP Purge Requests Forwarded               : 0
NHRP Purge Replies Received                 : 0
NHRP Purge Replies Transmitted              : 0
NHRP Purge Replies Forwarded                : 0
NHRP/MPOA Packets Dropped                   : 0
```

Entering **stats -e 1.2** from the atm/mps subsystem displays.

```
79:ASN-9000:atm/mps# stats -e 1.2
MPOA Server Errors on Segment 1.2
---------------------------------------------------------------------------
Unrecognized Extension Errors Received         : 0
Unrecognized Extension Errors Transmitted      : 0
Loop Detection Errors Received                  : 0
Loop Detection Errors Transmitted               : 0
Protocol Address Unreachable Errors Received    : 0
Protocol Address Unreachable Errors Transmitted : 0
Protocol Error Errors Received                  : 0
Protocol Error Errors Transmitted               : 0
SDU Size Exceed Errors Received                 : 0
SDU Size Exceed Errors Transmitted              : 0
Invalid Extension Errors Received               : 0
Invalid Extension Errors Transmitted            : 0
Auth Failure Errors Received                    : 0
Auth Failure Errors Transmitted                 : 0
Hop Count Exceed Errors Received                : 0
Hop Count Exceed Errors Transmitted             : 0
80:ASN-9000:atm/mps#
```

3.6.12 Cache

The **cache** command displays MPS cache entries for a given MPS segment, a list of segments or all configured MPS segments. The syntax for the command is:

cache [show] <seglist>|all

<seglist>all Specifies the segment, list of segments, or all segments for which to display trace settings.

Entering **cache 1.2** displays the MPS cache for segment 1.2.

```
79:ASN-9000:atm/mps# stats -e 1.2
IP Address     Mask           MPS Address  Type      ATM Address
198.29.21.23   255.255.255.0  198.29.21.67 Imposed   47.0005.80FFE1000000F21A00FB.0020481A02FB.00
198.29.21.53   255.255.255.0  198.29.21.67 Imposed   47.0005.80FFE1000000F21A00FB.0020481A02FB.00
198.29.41.23   255.255.255.0  198.29.21.67 Resolved  47.0005.80FFE1000000F21A00FB.0020481A02FB.00
79:ASN-9000:atm/mps#
```

where

IP Address Displays the destination IP address.

Mask Displays the destination IP address mask.

Type Displays the entry type that describes the cause of this cache entry.

ATM Address Displays the destination ATM address.

3.7 NHS Commands

Next Hop Server (NHS) is an MPOA service that performs address registration and resolution for shortcut or next-hop connectivity support. The NHS keeps a database of registered addresses, which clients query for ATM address resolution requests. If the database contains the address information, the server sends back the response with the corresponding ATM address. If the database does not contain the address information, the request is forwarded to other next hop servers. This section describes the commands used to configure and control NHS interfaces.

3.7.1 Interface

The **`it|interface`** command configures an NHS on a segment. A PowerCell segment also needs to be configured with IP-Over-NBMA (ION) protocol prior to configuring the NHS over it. The syntax for this command is:

```
it|interface add <segment> <sel-byte>
it|interface enl|enable <seglist>|all
it|interface dis|disable <seglist>|all
it|interface del|delete <segment>
```

where

add Adds a NHS interface on the specified <segment>. The optional selector byte can be provided for the NHS to use to control the ATM address. During creation, this selector byte is saved and used during the enable.

Before adding a segment to the NHS interface the following checks are made:

- The specified segment belongs to an ATM card
- The protocol for the specified segment is ION

del | delete Deletes an NHS interface on the specified <segment>

enl | enable Enables the NHS interface on the specified <segment-list> or all

dis | disable Disables the NHS interface on the specified <segment-list> or all

The following example adds and then enables a NHS interface on segment 1.3 with a selector byte of 0x84.

```
107:ASN-9000:atm/nhs# it add 1.3 84
NHS Added on Segment 1.3
108:ASN-9000:atm/nhs# it enl 1.3
109:ASN-9000:atm/nhs#
```

3.7.2 Configuration

The **`config`** command displays the NHS configuration for NHS interfaces configured on the specified segment, segment list, or all segments. The syntax for the command is:

```
config [show] <seglist>|all
```

Entering **`config 1.3`** displays the NHS configuration on segment 1.3. In this example a vlan was created containing ip address 144.125.75.33 on segment 1.3.

```
102:ASN-9000:config 1.3

NHS Configuration on segment: 1.3
---------------------------------------------------------------------------
Current status   : Down
IP address       : 144.125.75.33
ATM address      : (null)
Auth type        : none
Current clients  : 0
Max clients      : 1000
103:ASN-9000:atm/nhs#
```

where

Current Status Specifies whether the NHS is up or down.

IP address Displays the IP address configured on this segment. If an IP address is not configured, the NHS can not be operational. Use the **ip vlan** command to setup a vlan including this segment and then add an ip interface (**it add**).

ATM address The control ATM address for this NHS.

Auth type Indicates the type of authentication used by the NHS for NHRP protocol messages. Currently, this value is none.

Current clients Current number of registered clients.

Max clients Maximum number of clients allowed.

3.7.3 Statistics

The **stats** command displays NHS interface statistics of the specified NHS interfaces on the segment, segment list, or all segments. The use of the -e option displays only the error statistics of the specified segment(s). The syntax for this command is:

```
stats [show] [-e] <seglist>|all
stats clear <seglist>|all
```

Issuing **`stats all`** displays statistics for all segments configured with a NHS. The command below displays statistics for segment 1.3:

```
40:ASN-9000:atm/nhs# stats 1.3
NHRP stats on segment: 1.3
---------------------------------------------------------------------------
Resolution Requests Received             : 0
Resolution Requests Forwarded            : 0
Resolution Reply Acks Sent               : 0
Resolution Reply No Binding Naks Sent    : 0
Resolution Reply Not Unique Naks Sent    : 0
Resolution Replies Forwarded             : 0
Registration Requests Received           : 0
Registration Requests Forwarded          : 0
Registration Reply Acks Sent             : 0
Registration Reply Cant Serve Naks Sent  : 0
Registration Reply Overflow Naks Sent    : 0
Registration Reply Already Reg Naks Sent : 0
Registration Replies Forwarded           : 0
Purge Requests Received                  : 0
Purge Requests Sent                      : 0
Purge Requests Forwarded                 : 0
Purge Replies  Received                  : 0
Purge Replies  Sent                      : 0
Purge Replies  Forwarded                 : 0
Packets Dropped                          : 0
41:ASN-9000:atm/nhs#
```

Issuing **`stats -e all`** displays the error statistics on all segment NHS configured segments.

```
41:ASN-9000:atm/nhs# stats -e all
NHRP errors on segment: 1.3
---------------------------------------------------------------------------
Unrecongnized Extension Errors Received         : 0
Unrecongnized Extension Errors Transmitted      : 0
Subnet ID Mismatch Errors Received              : 0
Subnet ID Mismatch Errors Transmitted           : 0
Loop detection errors Received                  : 0
Loop Detection Errors Transmitted               : 0
Protocol Address Unreachable Errors Received    : 0
Protocol Address Unreachable Errors Transmitted : 0
Protocol Error Errors Received                  : 0
Protocol Error Errors Transmitted               : 0
SDU Size Exceed Errors Received                 : 0
SDU Size Exceed Errors Transmitted              : 0
Invalid Extension Errors Received               : 0
Invalid Extension Errors Transmitted            : 0
Auth Failure Errors Received                    : 0
Auth Failure Errors Transmitted                 : 0
Hop Count Exceed Errors Received                : 0

Hop Count Exceed Errors Transmitted             : 0
Fwd Error Indications                           : 0
42:ASN-9000:atm/nhs#
```

3.7.4 Cache

The **cache** command is used to add, delete or display cache entries of configured NHS interfaces on a specified segment, segment list, or all segments. The syntax of this command is:

```
cache [show] <seglist>|all
cache add <segment> <dest-ipaddr>[/<prefixlen>|<mask>]
[nhs[addr] <ipaddr>] atm[addr] <dest-atmaddr>
cache del|delete <segment> <dest-ipaddr>[/<prefixlen>|<mask>]
```

where

[show]

Displays the cache entries for the NHS interfaces configured on the segments specified in the <*segment-list*>, or all.

add <*segment*> <*dest-ipaddr*> [/<prefixlen> | <*mask*>] [nhs[addr] <*ipaddr*>] atm[addr] <*atmaddr*>

Adds a static entry to the cache of the specified segment. nhsaddr is optional if the destination is a host route.

segment the segment on which this NHS interface is configured.

ipaddr destination IP address / prefix length of the mask.

nhsaddr next hop address. For host routes this is the same as the destination address.

atmaddr corresponding atm address of the destination.

delete <*segment*> <*ipaddr*> [/ <*prefixlen*> | <*mask*>]

Deletes an entry from the cache of the specified segment.

segment the segment on which this NHS interface is configured

ipaddr destination IP address / prefix length of the mask.

3.7.5 Trace Settings

The **tracesettings|tr** command displays the trace settings configured on specified NHS interfaces or all NHS interfaces. The syntax for this command is:

```
[show] tracesettings|tr <seglist>|all
```

Entering **`tr 1.3`** displays the trace activity on segment 1.3.

```
46:ASN-9000:atm/nhs# tr 1.3
Entity     Action   Level    Enabled classes
---------------------------------------------------------------------------
NHS :
           Print    (warning)gen route txpkt rxpkt
           Log      (debug)  gen route txpkt rxpkt
nhs 1.3 :
           Print    (warning)gen route txpkt rxpkt
           Log      (debug)  gen route txpkt rxpkt
47:ASN-9000:atm/nhs#
```

3.7.6 Trace Level

The **tracelevel|trl** command sets trace levels to info, notice, or warning on specified NHS interfaces or all NHS interfaces. The syntax for this command is:

```
set tracelevel|trl level-name <seglist>|all
```

The example below sets the trace level to debug on segment 1.14:

```
56:ASN-9000:atm/nhs# set trl debug 1.14
```

Entering the **`tr`** command will display the trace level settings:

```
57:ASN-9000:atm/nhs# tr all
Entity       Action   Level        Enabled classes
---------------------------------------------------------------------------
NHS :
             Print    (warning)    gen route txpkt rxpkt
             Log      (debug)      gen route txpkt rxpkt
nhs 1.13 :
             Print    (info)       gen route txpkt rxpkt
             Log      (info)       gen route txpkt rxpkt
nhs 1.14 :
             Print    (debug)      gen route txpkt rxpkt
             Log      (debug)      gen route txpkt rxpkt
```

3.7.7 Trace Class

The **traceclass|trc** command enables or disables trace class of rxpkt, txpkt, route, timer or gen on specified or all NHS configured segments. The syntax for this command is:

```
enable traceclass|trc class-name <seglist>|all
disable traceclass|trc class-name <seglist>|all
```

The example below disables the route trace class on segment 1.14:

```
6:ASN-9000:atm/nhs# disable trc route1.14
```

3.8 FORE IP

This section describes ASN-9000 support for the FORE IP ATM protocol. FORE IP is a FORE Systems ATM protocol that emulates basic characteristics of an IP network.

NOTE

When setting up a new ATM network, FORE Systems recommends that LANE 1.0 or 2.0 be used to bridge or route between ATM and Ethernet. . Configure FORE IP on the PowerCell module only if the ATM network already uses FORE IP.

3.8.1 IP Characteristics Emulated by FORE IP

FORE IP emulates the following characteristics of the IP protocol:

- Address resolution using ARP.
- Dynamic connection establishment and teardown.
- Broadcast and multicast capability.

The FORE-IP implementation incorporates these services in software, using FORE System’s Simple Protocol for ATM Network Signalling (SPANS) and a Connectionless Service (CLS).

NOTE

The virtual interfaces created in FORE IP are based on IP and ATM addresses. The interfaces do not use MAC addresses to resolve destinations or routes. Because of this, all packets must be routed, not bridged, when destined for any other interfaces on the ASN-9000.

NOTE

Because SPANS addressing does not have a selector byte that can be assigned to multiple IP addresses on 1 segment, only assign FORE IP as the protocol for 1 segment of a given PowerCell.

3.8.2 The PowerCell Module and FORE IP

FORE IP networks contain the following components:

Simple Protocol for ATM Network Signalling (SPANS)
FORE System’s proprietary signaling protocol for use in ATM local-area networks. SPANs signaling occurs over VPI/VCI 0,15.

Connectionless Service (CLS)
A service that provides transport of connectionless traffic (IP broadcasts, OSPF, RIP, ARP requests and ARP responses) through an ATM network. An ATM network contains only one CLS. Connectionless traffic, including FORE IP traffic, is forwarded to and from the CLS using the well-known VPI/VCI pair 0,14.

NOTE

All FORE IP switched virtual circuits established using SPANs signaling are unidirectional. Each FORE IP connection requires an inbound VC and an outbound VC.

3.8.2.1 ARP Requests and Responses

ARP requests and responses are sent over a connectionless service, conforming with RFC826. The HARDWARE Type value in the ARP packet is set to 4040 (hex). The protocol type is set to 0800 (the Ethernet Type for IP packets). The hardware addresses for FORE in the ARP packets are the 8-byte SPANS ATM addresses.

3.8.2.2 IP Broadcasts

IP broadcast packets are dealt with in the same manner as ARP packets are—over the predefined VPI/VCI pair of 0,14. 0,14 is the VPI/VCI pair used for the CLS.

3.8.2.3 Point-to-Point IP Packets

Point-to-point IP packets are connection-oriented in nature; therefore, virtual circuits between IP hosts or ATM switches must be established. FORE-IP provides dynamic connection establishment using SPANS. For an existing connection between two IP hosts, IP packets are forwarded out the appropriate virtual circuit using the correct AAL type. If a connection does not exist, SPANS establishes a new connection.

3.8.2.4 IP Multicast

Point-to-multipoint connections are used for supporting IP multicast traffic over an ATM network, such that IP multicast packets can be transmitted from one source to multiple destinations. These point-to-multipoint connections are created using SPANS group addresses. An end station must first be added to the point-to-multipoint connection for the particular IP multicast group before the end station can receive IP multicast packets. The end station joins the multicast group by opening a point-to-multipoint connection to the group. IP Multicasting is supported by hardware point-to-multipoint connections on FORE Systems products; therefore, no special multicast processing is needed to service such multicast packets.

3.8.2.5 Configuring the ATM Switch for FORE IP

The PowerCell module supports FORE IP. FORE IP uses SPANS 1.0 to dynamically establish and teardown VCs .

NOTE

Disregard the procedures in this section if SPANS 1.0 on the ATM switch ports has already been configured for AAL5. Or you do not plan to use the FORE IP protocol.

(Segments for FORE IP can be reserved without actually enabling the protocol. If FORE IP segments are reserved in this way, configure the ATM switch before enabling FORE IP on the ASN-9000. If these configuration steps are not performed, FORE IP fails to operate when the protocol is enabled on the reserved segments.)

In order for the ATM switch and the ASN-9000 to communicate using FORE IP, both devices must be using the same ATM Adaptation Layer (AAL). For SPANS, the PowerCell module uses AAL5. However, by default, FORE Systems ATM switches support AAL3/4.

For the PowerCell module to communicate with FORE Systems ATM switches using FORE IP, SPANS AAL configuration on the port that directly connects the PowerCell module and the ATM switch must be changed to AAL5. For more information about SPANS AAL configuration, see your ATM switch manual.

3.8.3 Configuring for FORE IP

To use the PowerCell module for routing in a FORE IP network, perform the following configuration tasks:

- Configure one or more IP interfaces on the FORE IP segment (if not already done). For more information about adding an IP interface, see the IP Chapter.
- Set the ATM protocol type to FORE IP.
- Enable IP forwarding on an ASN-9000.
- Enable FORE IP on the segments configured for FORE IP.

NOTE

The IP address assigned to a FORE IP segment on the ASN-9000 can not be used on any other segment.

3.8.3.0.1 Configuring and Enabling FORE IP on a PowerCell Segment

Follow the procedure outlined below to configure FORE IP.

1. Before you can configure IP interfaces on the FORE IP segment, you must first the set the ATM protocol type to FORE IP using the following syntax:

```
proto sset f <seglist>
```

proto Specifies the protocol to be used on the segment

sset Sets the protocol to the specified segment

f Signifies the FORE IP option in the ATM subsystem

The example below shows the command:

```
52:ASN-9000:atm# atm sset proto f 1.5
```

2. Before you can enable FORE IP on the ASN-9000, you must first set up a vlan and configure one or more IP interfaces on the FORE IP segment . This is done in the IP subsystem. (For more detailed information about syntax for adding an IP interface, see the IP chapter in this manual.) To add a vlan, use the following command syntax:

```
vlan add <vlanid> <seglist>
```

The following example shows **the vlan** add command:

```
50:ASN-9000:atm# ip vlan add marketing 1.5
vlan marketing with segments 1.5 added
```

3. To configure IP interface on the FORE IP segments, issue the following command syntax:

```
it add <vlanid> <IP address>
```

The following is an example of the **it add** command:

```
51:ASN-9000:atm# ip it add marketing 124.123.12.12
Vlan marketing, Addr 124.123.12.12, Subnet mask 255.0.0.0,  type bcast Added
```

4. In the ATM subsystem enable FORE IP on the PowerCell segment with the following command syntax:

```
senable/sdisable <seglist>
```

<seglist> Specifies the segments on which FORE IP is to be enabled

Following is an example of this command:

```
54:ASN-9000:atm/foreip# senable 1.5
```

5. To View the configuration, use the **`config all`** command:

```
25:ASN-9000:atm# config all
Segment   Protocol     State      Rate Group
---------- --------------------- ----------
1.1       none         Disabled   1
1.2       lane         Enabled    1
1.3       none         Disabled   1
1.4       none         Disabled   1
1.5       foreip       Enabled    1
1.6       none         Disabled   1
```

3.8.3.0.2 Disabling FORE IP

The following example shows how to disable FORE IP on a segment:

```
7:ASN-9000:atm/foreip# sdisable 1.5
```

3.8.3.0.3 Displaying the IP Interfaces

To view the configured IP interfaces and their status on a particular segment or list of segments, enter the **ip it** command:

```
58:ASN-9000:atm# ip it
Vlan      Interface Addr Subnet Mask  Type  Neighbor Addr    MTU     Oper
forip     123.123.12.12  255.0.0.0    bcast  --------------- 1500    down
marketing 124.123.12.12  255.0.0.0    bcast ---------------  1500    down
test      146.111.111.22 255.255.0.0  bcast ---------------  1500    down
test1     147.11.22.33   255.255.0.0  bcast  --------------- 1500    down
techpubs  169.144.86.54  255.255.0.0  bcast ---------------  1500    up
IP Interface Count: 7
```

3.8.3.0.4 Displaying the Outbound Segment’s Cache

The ASN-9000 FORE IP software maintains a cache of the outbound VCs for each PowerCell segment configured for FORE IP. The FORE IP cache maps the IP address of a switch or end station to its corresponding SPANS address. The IP addresses can be for local or remote switches or end stations reachable through the corresponding SPANS addresses.

To display the FORE IP cache for a PowerCell segment, issue the following command:

`cache [show]` *`<seglist>`*

<seglist> Specifies the PowerCell segment to display the FORE IP cache.

Following is an example of the information displayed by this command. In this example, the FORE IP cache for segment 1.7 is displayed.

```
10:ASN-9000:atm/foreip# cache show 1.7
IP Address       Out VC   SPANS Address
----------       ------   -------------
134.163.20.3     121      00-00-00-01-f2-1a-23-bd
```

The fields in this display show the following information:

IP Address Indicates the destination IP address of the switch or the remote FORE IP workstation attached to the PowerCell segment.

Out VC Indicates the virtual channel identifier of the outbound VC. This VC is established by SPANS and is used by the PowerCell module to send FORE IP traffic from the PowerCell segment to the ATM switch or end station.

SPANS Address Indicates the destination SPANS address of the remote FORE IP workstation directly attached to the PowerCell segment.

In the following example, a single VC and SPANS address are associated with multiple IP addresses. This type of display is typical in configurations where multiple FORE IP end stations or ATM switches can be reached through the ATM switch attached to a PowerCell segment.

```
10:ASN-9000:atm/foreip# show cache 1.7
IP Address         Out VC    SPANS Address
----------         ------    -------------
134.163.20.3       121       00-00-00-01-f2-1a-23-bd
134.163.20.4       121       00-00-00-01-f2-1a-23-bd
134.163.20.5       121       00-00-00-01-f2-1a-23-bd
134.163.20.6       121       00-00-00-01-f2-1a-23-bd
```

In the example above, four different IP addresses can be reached through VC 121 and SPANS address 00-00-00-01-f2-1a-23-bd. The ATM switch associated with the SPANS address is locally attached. The IP addresses can be for the ATM switch or end station itself, or for other ATM switches or end stations that can be reached through the ATM switch associated with the SPANS address.

3.8.3.0.5 Displaying FORE IP Statistics

The ASN-9000 maintains FORE IP statistics for each PowerCell segment enabled for FORE IP. To display the FORE IP statistics for a PowerCell segment, issue the following command:

stats [show] *<seglist>*

<seglist> Specifies the PowerCell segment to display the FORE IP statistics.

Following is an example of the information displayed by this command. In this example, the FORE IP cache for segment 1.7 is displayed.

```
10:ASN-9000:atm/foreip# stats show 1.7
FORE-IP packet statistics for segment 1.7
   Total Pkts sent:               394
   Total ARP Pkts sent to CLS:    5
   Total BMCAST Pkts sent to CLS: 0
   Total Unicast Pkts sent:       389
   Total Pkts received:           426
   Total ARP Pkts received:       5
   Total BMCAST Pkts received:    0
   Total Unicast received:        421
   Total Pkts dropped:            0
   Total Pkts not sent:           0
   Total Pkts forwarded to PE:    389
   Total Pkts with bad length:    0
```

The fields in this display show the following information:

Total Pkts sent Indicates the total number of FORE IP packets sent on this PowerCell segment.

Total ARP Pkts sent to CLS Indicates the total number of FORE IP ARP packets sent by this PowerCell segment to the FORE IP CLS.

Total BMCAST Pkts sent to CLS Indicates the total number of FORE IP broadcast or multicast packets sent by this PowerCell segment to the FORE IP CLS.

Total Unicast Pkts sent Indicates the total number of FORE IP unicast packets sent by this PowerCell segment.

Total Pkts received Indicates the total number of FORE IP packets received on this PowerCell segment.

Total ARP Pkts received Indicates the total number of FORE IP ARP packets received on this PowerCell segment from the FORE IP CLS.

Total BMCAST Pkts received Indicates the total number of FORE IP broadcast or multicast packets received on this PowerCell segment from the FORE IP CLS.

Total Unicast Pkts received Indicates the total number of FORE IP unicast packets received on this PowerCell segment.

Total Pkts dropped Indicates the total number of packets dropped by this PowerCell segment.

Total Pkts not sent Indicates the total number of packets that were not sent by this PowerCell segment.

Total Pkts forwarded to PE Indicates the total number of FORE IP packets forwarded to the Packet Engine by this PowerCell segment. After receiving the packets from the PowerCell module, the Packet Engine discards or forwards the packets as needed.

Total Pkts with bad length Indicates the total number of FORE IP packets that did not have the correct length.

3.8.3.0.6 Clearing FORE IP Statistics

To clear FORE IP Statistics for a PowerCell segment, issue the following command:

```
stats clear <seglist>
```

After clearing the FORE IP statistics, the PowerCell module begins accumulating new statistics.

3.8.3.1 Configuring a FORE IP Network for Failover

A FORE IP network can be configured so that network connections are sustained if a failure occurs in one of the links between the PowerCell modules and ATM switches. To configure for failover, at least two PowerCell systems containing PowerCell modules and at least two ATM switches are required. Figure 3.19 shows an example of a FORE IP failover configuration.

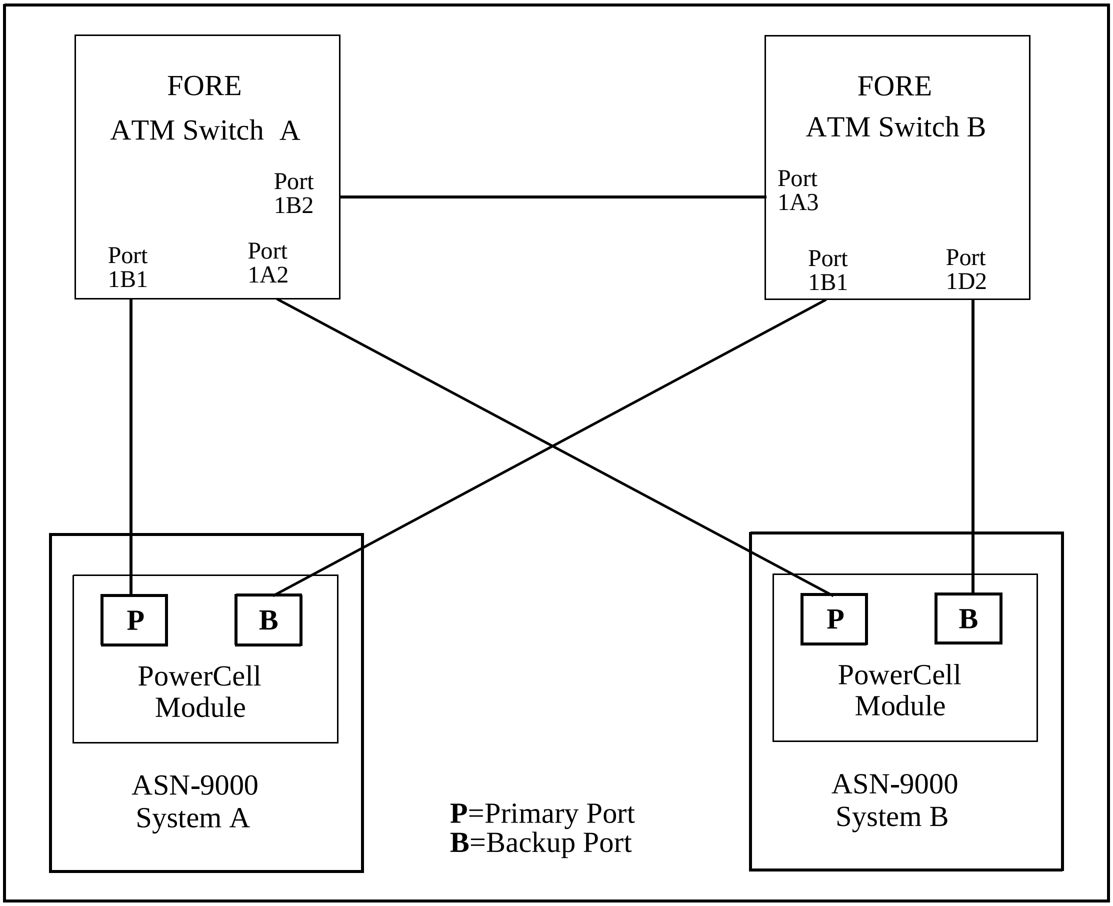


Figure 3.19 - Failover Configuration for FORE IP

To set up the configuration shown in Figure 3.19:

1. Connect the ASN-9000 modules to the ATM switches as shown in Figure 3.19.
2. If not already done, configure the FORE IP ports on the ATM switches to use interface `asx0`.

3. If not already done, enable FORE IP on the PowerCell segments for the FORE IP connections with the ASX switches.

NOTE It is not important which ports on the ATM switches the Fiber or UTP cables are plugged into, as long as the primary and backup ports on the PowerCell are connected to the correct switches as shown in Figure 3.19.

3.9 Classical IP over ATM

This section describes how the ASN-9000 supports Classical IP over ATM (CLIP). CLIP is an ATM Forum standard that allows IP datagrams and Address Resolution Protocol (ARP) requests and replies to be transmitted over ATM using ATM Adaptation Layer 5 (AAL5). CLIP is described in RFC 1577.

3.9.1 The PowerCell Module and CLIP

CLIP networks contain the following components:

Logical IP Subnet (LIS) A group of IP hosts or routers that are directly attached to an ATM switch and have the same IP network address, subnet address, and subnet mask. You can configure one LIS on each PowerCell segment you use for CLIP. After the CLIP network is initialized, individual members of the LIS are joined directly to other members by VCs (Virtual Channels). Hosts that are not members of the LIS can be reached only by using a LAN router.

ATM ARP server A device that can translate IP addresses into ATM addresses. When the ATM ARP server receives an ARP request from a host in an LIS, the ATM ARP server looks up the IP address supplied in the ARP request and returns the ATM address.

Virtual interfaces that are created in CLIP are based on IP and ATM addresses. The interfaces do not use MAC addresses to resolve destinations or routes. Because of this, all packets must be routed not bridged when destined for any other interfaces on the ASN-9000, including another LIS on the same PowerCell module.

Figure 3.13 shows an example of an ATM network using CLIP. Notice that each ATM host is a member of an LIS. In this example, the hosts are grouped into two LISs: 147.128.10.x and 147.128.20.x. The subnet mask used in the following example is 255.255.255.0.

NOTE

In release PH_FT 5.0.x, the ASN-9000 cannot be the CLIP ARP server.

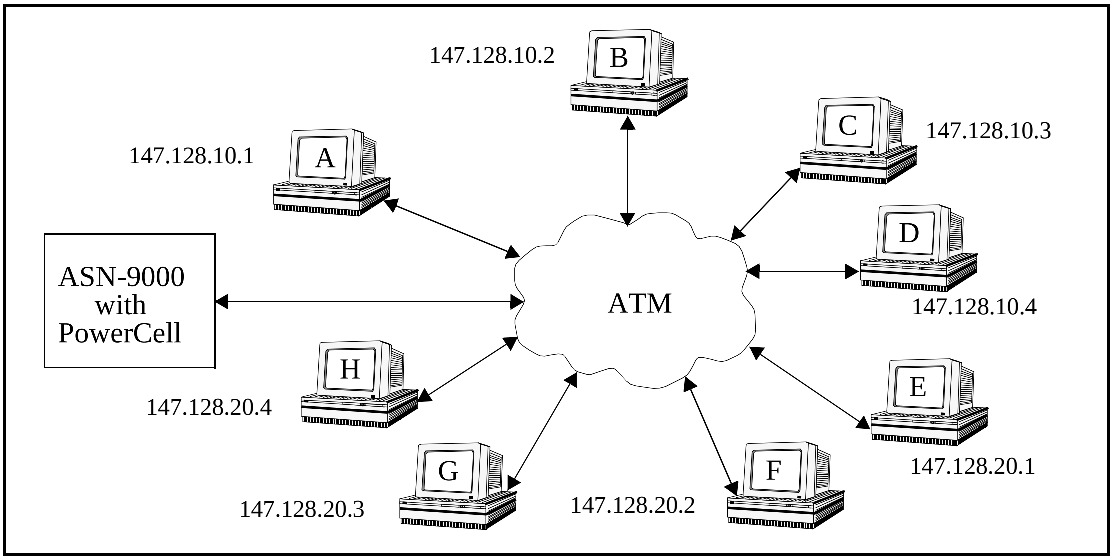


Figure 3.20 - CLIP Network

Figure 3.14 shows the same network from the ASN-9000 perspective. Each LIS is associated with its own PowerCell segment.

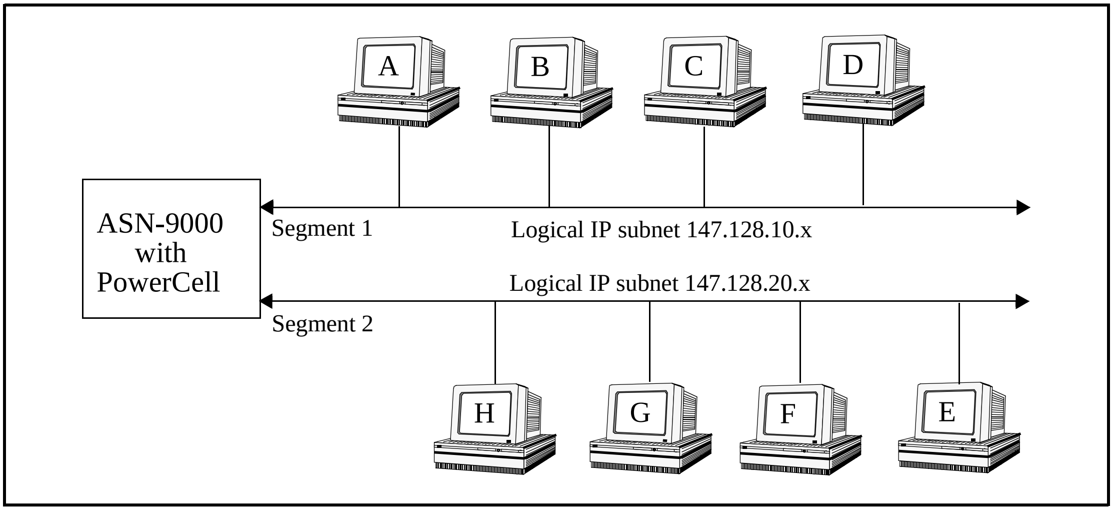

A
B
C
D
ASN-9000 with PowerCell
Segment 1
Logical IP subnet 147.128.10.x
Logical IP subnet 147.128.20.x
Segment 2
H
G
F
E

Figure 3.21 - CLIP Network Containing LISs

Figure3.15 shows two LISs connected to a PowerCell module, which is installed in a ASN-9000. Without a router to connect the LISs, the members of LISs cannot communicate with each other. The PowerCell module enables the LISs to communicate by routing IP traffic between the LISs.

3.9.1.1 SVC Support and Packet Encapsulation

The PowerCell module establishes connections between members of an LIS using SVCs. To establish an SVC, the PowerCell software uses Q.2931 signalling, as specified in the ATM Forum’s UNI 3.0 Specification.

After the PowerCell software establishes an SVC, the software encapsulates IP packets using IEEE 802.2 LLC/SNAP encapsulation and segments the packets into ATM cells using AAL5.

The default MTU is 9,180 bytes. When the SNAP header is added, the size becomes 9,188 bytes. The maximum IP datagram is 9180.

3.9.1.2 ATM ARP Support

CLIP uses ATM ARP and Inverse ATM ARP for address resolution within an LIS. ATM ARP is based on RFC 826 and Inverse ATM ARP is based on RFC 1293.

To configure a PowerCell segment to use an ATM ARP server in an LIS, specify the ATM Address of the ATM ARP server when configuring the PowerCell segment for CLIP. Perform this and other tasks using the **`clip sset`** command.

Each LIS must have one ATM ARP server. The same ATM ARP server can be shared across multiple LISs, but an LIS cannot contain two operational ARP servers. The ATM ARP server must have authoritative responsibility for resolving ATM ARP Requests from all IP nodes within the LIS.

When configuring the ATM ARP server, an IP address must be configured for each LIS the server supports.

When configuring an IP node for CLIP, the ATM address of the ATM ARP server must be specified in the LIS to which the node belongs. For example, when configuring a PowerCell segment for CLIP, one of the required configuration tasks is specifying the ATM address of the ATM ARP server.

3.9.1.2.1 ATM ARP Table Aging

ATM ARP entries in the ATM ARP table on the ATM ARP server are valid for a minimum of 20 minutes. Configure the aging interval for the ATM ARP entries on the ATM ARP server itself. See the documentation for your ATM ARP server for information.

After the aging interval on the server expires, the server generates an Inverse ARP Request on the open VC (if any) associated with the ARP entry.

- If the server receives an Inverse ARP Reply on a VC, the entry is updated and the aging timer starts over for the entry.
- If the server does not receive at least one Inverse ARP Reply, the server removes the entry from its ATM ARP table.

As a default, the ATM clients maintain ARP information for a duration of 15 minutes. The default time can be reconfigured with the **aa sset** command from the atm/clip subsystem.

To override the default arp aging interval, use the following command syntax:

```
atmarp-addr|aa sset <arpsvr-atm-addr> <seglist>
```

Below is an example of this command:

```
5:ASN-9000:atm/clip# aa sset 500 1.4
```

3.9.1.3 MTU Size

The default MTU size for IP members operating over the ATM network is 9180 bytes. The LLC/SNAP header is 8 bytes, therefore the default ATM AAL5 protocol data unit size is 9188 bytes. In Classical IP subnets, values other than the default can be used if all members in the LIS have been configured to use the non-default value.

If a Classical IP packet is locally forwarded by the PowerCell module from one LIS to another LIS attached to the module, the packet is forwarded without being fragmented. However, if the PowerCell module sends the packet to the Packet Engine for processing (for example, if the packet is destined for a segment on another module in the ASN-9000), the module fragments the packet before sending it to the Packet Engine. The fragments can be a maximum of 4060 bytes long.

3.9.2 Configuring a PowerCell Segment for Classical IP

To use a PowerCell segment for Classical IP routing in an ATM network, perform the following configuration tasks for each segment:

- Configure a vlan.
- Configure an IP interface on the segment (if not already done). Use the **`ip interface add`** command in the IP subsystem. When configuring the IP interface, select either of the two the interface types supported in ATM/CLIP: Non-Broadcast Multiple Access (NBMA) or Point-to-Point. These interface types provide the ability to run IP routing protocols over non-broadcast interfaces. Neighbor address must be specified only for ptop type.
- Set the ATM protocol type to CLIP. (Use the **`sset protocol`** command.
- Specify the ATM address of the ATM ARP server attached to the segment and enable Classical IP on the segment. (Use the **`clip sset`** command.)

Enable CLIP on the segment using **`senable`** command.

The following sections describe how to perform these tasks.

3.9.2.1 Configuration Considerations

Before configuring the PowerCell module for CLIP, make sure the configuration plans are not affected by the following considerations:

- Only one IP interface can be configured on a PowerCell segment enabled for Classical IP.

NOTE IP routes must be statically configured on a PowerCell segment enabled for Classical IP.

- Broadcast traffic, such as RIP or OSPF, is not supported, as there is no mechanism in place to distribute broadcast packets. If the segments being configured require the ability to send and receive broadcast traffic, use LANE 1.0 or 2.0 on the segments. The two interface types supported in CLIP are Non-Broadcast Multiple Access (NBMA) or Point-to-Point. These interface types provide the ability to run

IP routing protocols over non-broadcast interfaces. When NBMA with neighbors is configured for an interface (either RIP or OSPF), RIP / OSPF updates are learned from neighbors.

- Layer-3 VLANs are not supported on PowerCell segments configured for CLIP. To configure a Layer-3 VLAN on multiple PowerCell segments, use LANE 1.0, 2.0, or 1483 LANE Frame PVCs on the segments.
- By default a port gets removed from a bridge group when configured for CLIP.

3.9.2.2 Configuring and Enabling a Segment for Classical IP on ATM

To configure and enable CLIP on a segment, the following parameters must be configured:

1. Configure a vlan by enter the following command from the IP subsystem:

```
vlan add <vlanid><segment>
```

Here's an example of the **vlan add** command:

```
3:ASN-9000:ip# vlan add QA 1.23
vlan QA with segments 1.23 added
```

2. Add an interface IP address, using the following command syntax form the IP subsystem:

```
it add <vlanid><IP address>
```

Here's an example of the **it add** command:

```
20: ASN-9000:ip# it add QA 167.122.98.11
Vlan QA, Addr 167.122.98.11, Subnet mask 255.255.0.0,  type bcast Added
```

3. Configure the PowerCell segment to use CLIP by telnetting into or connect to the ASN-9000 through the TTY interface. From the **`atm`** subsystem, set the protocol to CLIP using the following command syntax:

```
proto sset c <seglist>
```

proto	Specifies the protocol set on the segment
c	Specifies classical IP protocol
segment	Specifies the segment to be configured

Here is an example of the **proto sset** command:

```
44:ASN-9000:atm# proto sset c 1.23
```

4. Specify the ATM address of the ATM ARP server. The syntax for this command is:

atmarpserver-addr|as sset *<arpsvr-atm-addr> <segment>*

<arpsvr-atm-addr> Specifies the ATM address of the ATM ARP server. Specify the address in NSAP format.

<segment> Specifies the PowerCell segment being configured for Classical IP.

Here is an example of the **atmarp sset** command. The command specifies the ATM address of the ARP server on the LIS.

```
8:ASN-9000:atm/clip# as sset  45.0005.80.ffe100.0000.f215.1490.00-00-ef-01-ab-cd. 04
1.23
```

5. To set ARP-aging on the ARP server, use the **arp-aging sset** command. The syntax for this command is:

arp-aging|aa sset *<seconds> <seglist>*

<seconds> Specifies the maximum time that an ATM ARP entry is kept without being used. The minimum value that can be set is **1200** seconds and the maximum value is **3600** seconds. The default is **1200** seconds.

<seglist> Specifies the segment being configured for ARP-aging on the segment.

Here is an example of the **arp-aging sset** command.

```
10:ASN-9000:atm# aa sset 500 1.23
```

6. To set ARP-aging timeout on the ARP server, use the **arp-conn-timeout sset** command. The syntax for this command is:

arp-conn-timeout|at sset *<seconds> <seglist>*

<seconds> Specifies the time to wait when connecting to an ARP server to detect if the attempt failed. The ASN-9000 automatically tries to reconnect if the attempt failed. The minimum value that can be set is **5** seconds and the maximum value is **60** seconds. The default is **10** seconds.

<seglist> Specifies the segment being configured for ARP-aging connection timeout.

Following is an example of the **arp-conn-timeout sset** command:

```
27:ASN-9000:atm/clip# at sset 30 1.23
```

7. From the atm/clip subsystem, enable CLIP on the segment(s) with the following command:

sensable <seglist>

Here's an example of the **senable** command:

```
30:ASN-9000:atm/clip# senable 1.23
```

3.9.2.3 Displaying the Classical IP Configuration

To verify the Classical IP configuration of a PowerCell segment, issue the following command:

config [show] [local|l] *<seglist>***|all**

[local|l] Specifies the locally stored information on the packet engine.

<seglist>|all Specifies the PowerCell segment(s) to display the Classical IP configuration. Specify a single segment number, a comma-separated list of segments, or a hyphen-separated range of segments. If **all** is specified, configuration information is displayed for all the PowerCell segments that are configured for Classical IP.

The **config** command shows the following display:

```
31:ASN-9000:atm/clip# config 1.23
Displaying information from the ATM Card for segment 1.23
Show remote cfg for segment 1.23
Classical-IP-Over-ATM Configuration for segment 1.23
--------------------------------------------------
IP Address:              167.122.98.11
Interface ATM Addr:      47.0005.80ff.e100.0000.f21a.1bdd.0000ef039ab1.16
ATM ARP Svr Addr:        47.0002.80ff.e100.0003.f211.1430.0000ef01abcd.04
Admin State:             Enabled
Physical State:          Link Up
Oper State:              Initializing
IP I/F Status:           IP I/F Configured
ARP Age (secs):          500
ARP Conn timout (secs):  30
```

3.9.2.4 Displaying Classical IP Statistics

To display Classical IP statistics for a PowerCell segment, issue the following command:

```
stats [show] <seglist>|all
```

<seglist>|all Specifies the PowerCell segment(s) to display Classical IP statistics. Specify a single segment number, a comma-separated list of segments, or a hyphen-separated range of segments. If **`all`** is specified, statistics are displayed for all the PowerCell segments that are configured for Classical IP.

The **stats** command shows the following information:

```
16:ASN-9000:atm/clip# stats 1.5
Displaying statistics from the ATM Card for segment 1.5
Classical-IP-Over-ATM Statistics for segment 1.5
--------------------------------------------
Connection Fails:             0
Total Control Packets In:     0
Total Control Packets Out:    0
Arp Replies In:               0
Arp Replies Out:              0
Total Arp Replies :           0
Arp Requests In:              0
Arp Requests Out:             0
Deleted Arp Replies:          0
Unknown Arp Replies:          0
Total InARP Requests:         0
Total ARP NAKs:               0
Total bad ARP operations:     0
Total times CLIP restarted:   18
Unknown Packets received:     0
Unicast Data in:              0
Bad ip packets in:            0
Unicast Packets dropped:      0
Unicast packets forwarded:    0
```

The fields in this display show the following information:

Total Control Packets In Indicates the number of control packets received on the PowerCell segment(s).

Total Control Packets Out Indicates the number of control packets sent on the PowerCell segment(s).

ARP Replies In Indicates the number of ARP replies received from the LIS on the segment

ARP Replies Out Indicates the number of ARP replies sent on the segment.

Total ARP Replies Indicates the total number of ARP replies received and sent.

ARP Requests In Indicates the number of ARP requests received from the LIS on the segment.

ARP Requests Out Indicates the number of ARP requests sent to the LIS ARP server on the segment.

Deleted ARP Replies Indicates the number of deleted ARP replies on the segment.

Unknown ARP Replies Indicates the number or unknown ARP replies received on the segment.

Total Inverse ARP Requests Indicates the number of inverse ARP request received from the LIS ARP server on the segment.

Total ARP NAKs Indicates a total of NAKs received on the segment.

Total bad ARP Operations Indicates the number of failed ARP operations.

Total times CLIP Restarted Indicates the total time CLIP was re-initialized.

Unknown Packets Received Indicates packets received with unidentified protocol IDs.

Unicast Data In Indicates the number of unicast packets received on the PowerCell segment.

Bad IP Packets In Indicates the number of bad IP packets received on the segment.

Unicast Packets Dropped Indicates the number of unicast packets that were dropped by the PowerCell module rather than sent.

Unicast Packets Forwarded Indicates the number of unicast packets forwarded on the PowerCell segment.

3.9.2.4.1 Clearing Classical IP Statistics

To clear Classical IP statistics for a PowerCell segment, issue the following command:

```
stats clear <seglist>|all
```

<seglist>|all Specifies the PowerCell segment(s) to clear Classical IP statistics. Specify a single segment number, a comma-separated list of segments, or a hyphen-separated range of segments. If **all** is specified, statistics are cleared for all the PowerCell segments that are configured for Classical IP.

3.9.2.5 Removing a Classical IP Segment

To remove CLIP from a segment, disable the segment using the **sdisable** command and then remove the protocol form the segment using the **atm sset protocol** command. The **sdisable** command is described in section 3.9.2.5.

Following are some examples of the commands used to disable a CLIP segment. The first command disables the segment and removes it from use, and the second command removes the CLIP protocol from the segment. In this example, the terse form of the **sdisable** and **atm sset protocol** commands are used. Note that when a segment is disabled, it is not necessary to specify the ARP server on the LIS.

```
12:ASN-9000:atm/clip# sdisable 1.23
13:ASN-9000:atm# proto sset n 1.23
```

3.9.2.5.1 For Members of an LIS

The requirements for IP members (hosts, routers) operating in an ATM LIS configuration are as follows:

- All members have the same IP network number, subnet number, and subnet mask.
- All members within the LIS must be directly connected to the ATM network. Members outside of the LIS can be accessed only by a router .
- All members of the LIS must have a mechanism for resolving IP addresses into ATM addresses using ATM ARP. Members attached by SVCs must be able to resolve IP addresses into ATM addresses using Inverse ATM ARP.
- All members of the LIS must have a mechanism for resolving VCs into IP addresses using Inverse ATM ARP over PVCs.
- All members within the LIS must be able to communicate through ATM with all other members of the same LIS. That is, the VC topology underlying the interconnection among the members must have the ability to be fully meshed.

3.9.2.5.2 ATM Parameters for Classical IP

The following list identifies a set of ATM specific parameters that must be implemented on each IP node connected to the ATM network:

- The ATM address of the node. The address is resolved automatically by ILMI.
- The ATM address of the ATM ARP server in the LIS to which the node belongs. In an SVC environment (such as the one including the PowerCell module), ATM ARP requests are sent to this address for the resolution of target protocol addresses to target ATM addresses. The ATM ARP server must have authoritative responsibility for resolving the ATM ARP requests of all the IP nodes in the LIS.

3.10 Classical IP PVC over ATM

This section describes ASN-9000 support for CLIP PVC over ATM. Normally, ATM connections in a Classical IP environment are established dynamically using UNI 3.0 or 3.1. ARP, ILMI, and UNI 3.0 or 3.1 all work together when setting up an SVC. If a host or switch in an LIS does not support UNI 3.0, however, it is not possible to establish an SVC. In this case, a Classical IP PVC can be used for communication.

On each of the CLIP PVC ASN-9000 segments, the **`sset`** command is used to establish the PVC. An unused VCI must be chosen for each CLIP PVC ASN-9000 segment. PVCs using the chosen VCI must also be setup from each of the hosts to their connecting switch, and then on all of the switches between the two connecting switches.

NOTE

Both the incoming and outgoing connections are set up simultaneously on the host, but they must be set up individually on the switches. The same VCI is used by a host to send on the PVC as well as receive on the PVC. The IP datagrams are sent over the PVC using AAL5 with LLC/SNAP encapsulation.

Figure 3.16 shows an example of an ATM network using CLIP PVC. Notice that each ATM host is a member of an LIS. In this example, the hosts are grouped into two LISs: 147.128.10.x with PVCs 100-104 on segment 1 and 147.128.20.x with PVCs 201-205 on segment 2.

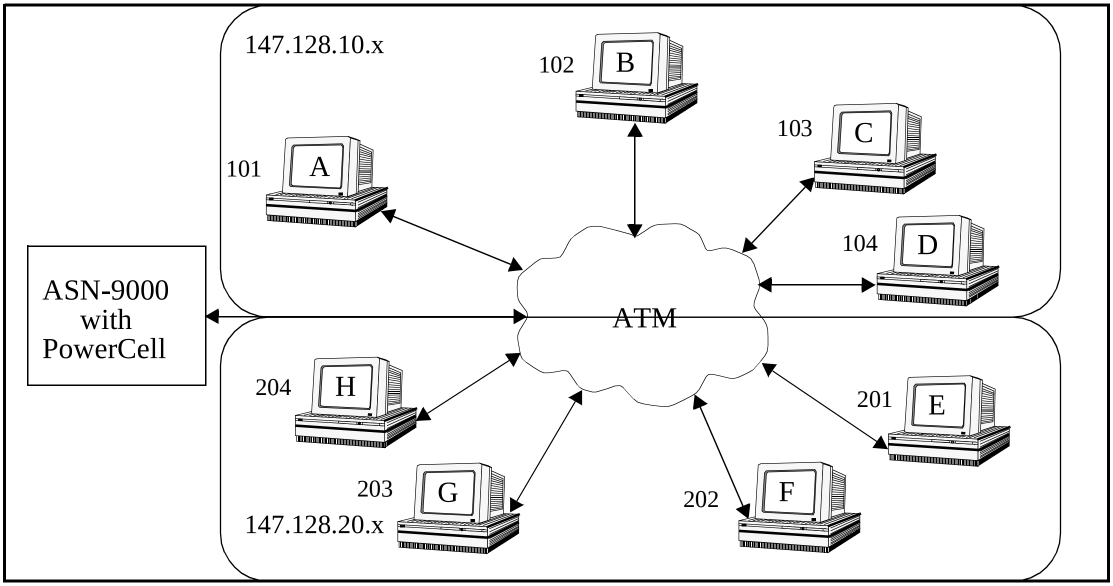


Figure 3.22 - CLIP PVC Network

Figure 3.17 shows the same network from the ASN-9000 perspective. Each LIS is associated with its own PowerCell segment.

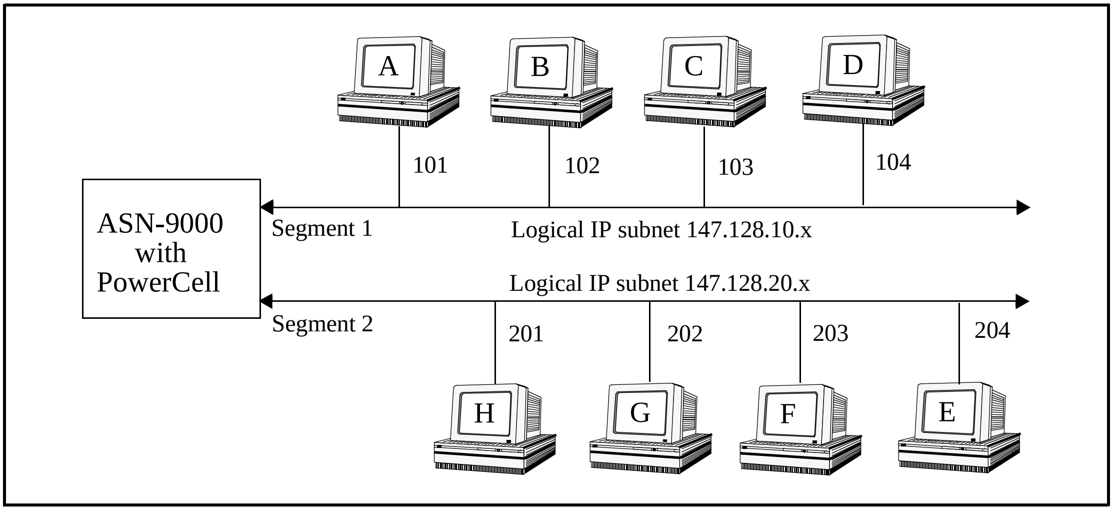


Figure 3.23 - CLIP PVC Network Containing LISs

Figure 3.17 show two LISs connected to a PowerCell module, which is installed in a ASN-9000. Without a router to connect the LISs, the members of LISs cannot communicate with each other. The PowerCell module enables the LISs to communicate by routing IP traffic between the LISs. From the ASN-9000 in Figure 3.17, segment 1 connects with PVC 101, with a logical IP subnet 147.128.10x, to station A, 102 to station B, and so on. Segment 2 connects with PVC 201, with a logical IP subnet 147.128.20x, to station H, 202 to station G, and so on.

3.10.1 PVC Support and Packet Encapsulation

The PowerCell module can establish connections between members of an LIS using PVCs (Permanent Virtual Circuits). After the PowerCell software establishes a PVC, the software encapsulates IP packets using IEEE 802.2 LLC/SNAP encapsulation, and segments the packets into ATM cells using AAL5.

The default MTU is 9,180 bytes. When the SNAP header is added, the size becomes 9,188 bytes. The maximum packet size is 9180. The same (MTU) size is used for all VCs in a LIS.

3.10.2 ATM ARP Support

CLIP PVC uses Inverse ATM ARP to resolve the IP addresses of the host at the other end of a VC. Inverse ATM ARP is based on RFC 1293.

To configure a PowerCell segment to use CLIP PVC specify the VC using the **`clippvc sset`** command. Section 3.8.2 explains how to configure CLIP PVC.

When configuring a PowerCell segment to support CLIP PVC, the segment must first be IP configured and routing must be enabled.

3.10.2.1 ATM ARP Table Aging

When the requesting station receives the Inverse ARP (InARP) reply, it may complete the ATM ARP table entry and use the provided address information. It is the responsibility of each IP station supporting PVCs to revalidate ATM ARP table entries as part of the aging process. The ASN-9000 responds in ARP. A host revalidates a PVC every 15 minutes by sending InARP requests over the PVC. If an InARP reply is not received, the revalidation fails, the PVC is marked invalid (as shown through the **`config`** command), and communication over the PVC is no longer possible.

Once a PVC is marked invalid, an attempt is made to validate the PVC before transmitting. Transmissions proceed only when validation succeeds. Client ATM ARP table entries are valid for a maximum of 15 minutes This default can be over-ridden using the **`aa sset`** command (discussed below)..

- If the PowerCell receives an Inverse ATM ARP Reply on a VC, the entry is updated and the aging timer starts over for the entry.
- If the PowerCell does not receive an Inverse ATM ARP Reply, the entry is marked invalid in the ARP cache.

As a default, the ATM clients maintain ARP information for a duration of 15 minutes. The default time can be reconfigured with the **`aa sset`** command from the atm/clip subsystem.

To override the default arp aging interval, use the following command syntax:

```
arp-aging|aa sset <seconds> <seglist>
```

<seconds> Specifies the maximum time that an ATM ARP entry is kept without being used. The minimum value that can be set is **1200** seconds and the maximum value is **3600** seconds. The default is **1200** seconds.

<seglist> Specifies the segment being configured for ARP-aging on the segment.

Here is an example of the **arp-aging sset** command.

```
10:ASN-9000:atm# aa sset 500 1.23
Okay
```

3.10.2.2 CLIP PVC ARP Display

Upon enabling a configured CLIP PVC segment on an ASN-9000, an Inverse ATM ARP request is sent out on the PVC. When a response is received from the peer member or station, the entry is put into the ATM ARP table of the PowerCell. The entries that are learned through the Inverse ATM ARP can be displayed using the **arp show** command. The syntax for this command is:

arp show *<seglist>*|**all**

<seglist>|all Displays the cache entries established through CLIP PVC on the specified segment.

The following are the results produced by this command. Displayed is an Inverse ATM ARP request sent on ASN-9000 segment 1.1 over PVC.

```
117:ASN-9000:atm/clippvc# arp all
Configured PVCs and state:
IP Address      PVC     Segment    State
------------    ---     -------    -----
147.128.10.1    300      1.1        VALID
```

IP Address Indicates the IP address of the peer member of the LIS that has responded to the Inverse ATM ARP request.

PVC Indicates the configured PVC.

Segment Displays the segment configured for CLIP PVC.

State Displays the state of the entry that responded to the Inverse ATM ARP request. This field is valid when the responding member for the Inverse ATM ARP request is a member of the LIS as the ASN-9000 segment configured for CLIP PVC.

The state is invalid if the peer LIS member does not respond to the Inverse ATM ARP request after the timeout period or if the member responding is not a member of the LIS.

The following example shows PowerCell segment 5.2 configured for CLIP PVC with an IP address of 100.1.1.2. The peer LIS member, with an IP address of 100.1.1.3 has responded to the Inverse ATM ARP request through an Inverse ATM ARP reply. The state of the PVC is "VALID."

```
117:ASN-9000:atm/clippvc# arp all
Configured PVCs and state:
IP Address      PVC    Segment    State
100.1.1.3       200      5.2      VALID
```

When a PowerCell segment is configured for CLIP PVC and a non-member of the LIS responds to his inverse ATM ARP request, the state will be “INVALID.”

In the following example, segment 5.3 is configured with an IP address of 200.1.1.2. A non-LIS member with an IP address of 200.1.2.3 has responded to an Inverse ATM ARP request through an Inverse ATM ARP reply.

```
117:ASN-9000:atm/clippvc# arp all
Configured PVCs and state:
IP Address     PVC    Segment State
------------   ---    ------- -------
200.1.2.3      500    5.3     INVALID
```

When a PowerCell segment is configured for CLIP PVC and a peer LIS member stops responding to an Inverse ATM ARP request after the revalidation interval is reached, the state is set to “INVALID.”

In the following example, segment 5.2 is configured with an IP address of 100.1.1.2. The peer member with an IP address of 100.1.1.3 has stopped responding to the Inverse ATM ARP request and the state is set to “INVALID.”

```
117:ASN-9000:atm/clippvc# arp show all
Configured PVCs and state:
IP Address     PVC    Segment    State
------------   ---    -------    -------
100.1.2.3      200    5.2        INVALID
```

3.10.3 MTU Size

If a Classical IP packet is locally forwarded by the PowerCell module from one LIS to another LIS attached to the same module, the packet is forwarded without being fragmented. However, if the PowerCell module sends the packet to the Packet Engine for processing (for example, if the packet is destined for a segment on another module in the ASN-9000), the module fragments the packet before sending it to the Packet Engine. The fragments can be a maximum of 4060 bytes long.

3.10.4 Configuration Considerations

Before configuring the PowerCell module for CLIP PVC, make sure the configuration plans are not affected by the following considerations:

- Only one IP interface can be configured on a PowerCell segment enabled for CLIP PVC.
- Broadcast traffic is not supported, as there is no mechanism in place to distribute broadcast packets. If the segments being configured require the ability to send and receive broadcast traffic, use LANE 1.0 or 2.0 on the segments.
- One IP interface can be configured on a PowerCell segment, for a maximum of 32 IP interfaces on a PowerCell module.
- Layer-3 VLANs are not supported on PowerCell segments configured for CLIP PVC. To configure a Layer-3 VLAN on multiple PowerCell segments, use LANE 1.0 or 2.0 on the segments.
- Do not include segments configured for CLIP PVC in ASN-9000 bridge (network) groups.

3.10.5 Configuring a PowerCell Segment for CLIP PVC

To use a PowerCell segment for CLIP PVC routing in an ATM network, perform the following configuration tasks for each segment:

- Configure a vlan.
- Configure an IP interface on the segment (if not already done). Use the **`ip interface add`** command in the IP subsystem. When configuring the IP interface, select either of the two the interface types supported in ATM/CLIP: Non-Broadcast Multiple Access (NBMA) or Point-to-Point. These interface types provide the ability to run IP routing protocols over non-broadcast interfaces. Neighbor address must be specified only for ptop type.
- Set the ATM protocol type to Classical-IP PVC. (Use the **`atm sset protocol`** command.)
- Specify the PVC and the revalidation interval period (If nothing is specified for the revalidation period, the default of 15 minutes is used).)
- Enable IP routing.

1. After adding an IP interface on the segment, configure the PowerCell virtual segment to use CLIP PVC using the **`atm sset protocol`** command. To configure the PowerCell segment to use CLIP PVC, telnet into or connect to the ASN-9000 through the TTY interface, change to the **atm** subsystem, and configure the desired segment using the following command:

   `sset proto[col]` *<proto>* **`<segment>`**

   <proto> Specifies the protocol to be used on a segment. To configure the PowerCell segment to use CLIP PVC issue the following:

   cp Specifies classical-ip-pvc

   <segment> Specifies the PowerCell segment being configured for CLIP PVC. Specify a single segment number, a comma-separated list of segments, or a hyphen-separated range of segments. If **`all`** is specified, all segments on all PowerCell modules in the chassis are configured to use the CLIP PVC protocol.

2. After configuring a PowerCell segment to use CLIP PVC, specify the PVC to be used on the segment using the following syntax:

   `sset <pvc><segment>[revalidation time>]`

   <pvc> Specifies the PVC to be used on the segment. Valid PVCs range between 32 and 1023.

   <segment> Specifies the PowerCell segment being configured for CLIP PVC.

   <revalidation time> Specifies the revalidation interval after sending an Inverse ATM ARP to validate the LIS member using this PVC. The Inverse ARP reply, received by the PVC configured PowerCell segment, discovers and keeps the IP address of the LIS member.

The first command in the example below sets the PVC on segment 1.22 and a revalidation time of 30 minutes. The second command ignores the revalidation interval by defaulting to 15 minutes.

```
23:ASN-9000:atm/clippvc# sset 300 1.22 30
```

3. Enable the CLIP PVC configuration with the **senable** command, using the following syntax:

`senable <seglist>`

Here's an example of this command:

```
25:ASN-9000:atm/clippvc# senable 1.22
```

3.10.5.1 Removing CLIP PVC from a Segment

To remove CLIP PVC from a segment, perform the following steps:

1. Disable the segment using the **`sdisable`** command

`sdisable <seglist>|all`

The example below shows the sdisable command:

```
1:ASN-9000:atm/clippvc# sdisable 1.22
Disabling the enabled PVCs segment 1.22
```

2. Remove all configured PVCs from a segment, using the **`pdelete`** command. The **`pdelete`** command is used to delete a single PVCs or all of the PVCs from a CLIP PVC segment. The command syntax is as follows:

`pdelete [<pvc><seglist>]|all`

In the example below, the pdelete command specifies the PVC and the segment.

21:ASN-9000:atm / clippvc# pdelete 300 1.22

The example below shows the use of this command:

```
21:ASN-9000:atm/clippvc# pdelete 300 1.22
```

3. Undefine the protocol from the atm subsystem using the **`proto sset none`** command. The syntax for this command is as follows:

```
proto sset n [<PVC><seglist>]|all
```

The example below shows the use of this command:

```
22:ASN-9000:atm/clippvc# atm proto sset n 300 1.22
```

3.10.5.1.1 For Members of an LIS

The requirements for IP members (hosts, routers) operating in an ATM LIS (Logical IP Subnet) configuration are as follows:

- All members have the same IP network number, subnet number, and subnet mask.
- All members within the LIS must be directly connected to the ATM network. Members outside of the LIS can be accessed only by a router.
- All members of the LIS must have a mechanism for resolving IP addresses into ATM addresses using ATM ARP.
- All members of the LIS must have a mechanism for resolving VCs into IP addresses using Inverse ATM ARP over PVCs.
- All members within the LIS must be able to communicate through ATM with all other members of the same LIS. That is, the VC topology underlying the interconnection among the members must have the ability to be fully meshed.

3.10.5.1.2 ATM Parameters for Classical IP

The following list identifies a set of ATM specific parameters that must be implemented on each IP node connected to the ATM network:

- The ATM address of the node. The address is resolved automatically using ILMI (Interim Link Management Interface).
- The PVC that is used to communicate to a peer member of the LIS. Inverse ATM ARP resolves the IP address of the peer member of the LIS.
- The revalidation interval used after the sent Inverse ARP requests update the peer LIS member's active or inactive state.

3.10.6 Displaying the CLIP PVC Configuration

The current CLIP PVC configuration can be displayed using the **`config [show]`** command. The following example shows the display produced by this command:

```
13:ASN-9000:atm/clippvc# config all
Configured PVC's and state
PVC     State  Segment Reval Time
---      -----  ------- ----------
300     dis    1.22      15
412     dis    1.22      30
```

PVC Specifies the PVC associated with the segment.

State Indicates whether CLIP PVC is enabled or disabled on this segment.

Segment Shows the segment running CLIP PVC.

Reval Time Specifies the revalidation interval period (If nothing is specified for the revalidation period, the default is 15 minutes).

3.10.6.1 Displaying and Clearing Statistics

The current CLIP PVC statistics can be displayed using the **`stats [show]`** command. Following is an example of the display produced by this command:

```
117:ASN-9000:atm/clippvc# stats 1.22
Displaying statistics from the ATM Card for segment 1.1
Classical-PVC-Over-ATM Statistics for segment 1.1
--------------------------------------------
Connection Fails:               0
Total Control Packets In:       0
Total Control Packets Out:      0
Arp Replies In:                 0
Arp Replies Out:                0
Total Arp Replies :             0
Arp Requests In:                0
Arp Requests Out:               0
Deleted Arp Replies:            0
Unknown Arp Replies:            0
Total InARP Requests:           0
Total ARP NAKs:                 0
Total bad ARP operations:       0
Total times CLIP restarted:     1
Unknown Packets received:       0
Unicast Data in:                17554
Bad ip packets in:              0
Unicast Packets dropped:        1
Unicast packets forwarded:      17553
```

Use the **`stats clear`** command to clear CLIP PVC over ATM statistics. All learned entries are removed, but static entries (created using the **`sset atmarp`** command) remain in the table. These must be removed manually using the **`pdelete`** command.

This command can be used to help restabilize the network after a host is moved from one segment to another. When there is activity on the network, the cleared entries quickly reappear in the ATM ARP table, and a host that has been moved will be relearned on its new segment.

A Classical IP PVC is removed on the host side with the **`pdelete`** command after disabling the PVC segment using the **`sdisable`** command. Both incoming and outgoing connections are removed simultaneously. The PVC must then be removed from each of the network switches involved.

CHAPTER 4 Digital Network Routing (DECnet)

The ASN-9000 contains a complete set of DECnet Phase IV routing software for use in DECnet networks. The routing engine works side-by-side with the Ethernet bridging software. With appropriate configuration, the ASN-9000 can be set up to perform DECnet routing on any segments.

This chapter assumes a familiarity with the basic requirements of DECnet networks and the DECnet protocol. For further information on this subject, refer to a DECnet guide, such as the DECnet Phase IV General Description, Order No. AA-N149A-TC, (Digital Equipment Corporation, 1982).

This chapter describes the commands and facilities of the DECnet subsystem. To set up the ASN-9000 for DECnet routing, the following steps must be performed:

1. Allocate memory for DECnet routing.
2. Assign the DECnet node ID using the **`node-id set`** command.
3. If the ASN-9000 is to be a Level-2 router, select it with the **`node-type set`** command.
4. Turn on DECnet routing with the **`dec enable`** command.
5. Enable DECnet routing on the desired segments with the **`dec penable`** command.

A large number of nodes may necessitate increasing the maximum limits for these parameters with the **`max-area-num set`** and **`max-node-num set`**commands. After setting up DECnet routing, check connectivity to hosts and other routers using the **`show`** and **`stats`** commands. After configuring DECnet, it is recommended that the configuration be saved using the **`savecfg`** <*file-name*> command. See the *ForeRunner ASN-9000 Hardware Reference Manual.*

4.1 Accessing the DECnet Subsystem

To access the dec subsystem, issue the following command from the runtime command prompt:

dec

```
dec subsystem:

adjacent|adj                         max-hops-to-area|mha
area                                 max-hops-to-node|mhn
block-size|bs                        max-node-num|mnn
cache                                max-routers|mr
cost|c                                max-visits
dec                                   node-type|nt
getmem                                node-id|nid
hello-time|ht                         priority|pri
max-adj-endnodes|mae                  route|rt
max-adj-routers|mar                   routing-status|rs
max-area-num|man                      stats
max-cost-to-area|mca                   update-time|ut
max-cost-to-node|mcn
```

4.1.1 Allocating Memory

Before using the dec subsystem, allocate memory for the subsystem by issuing the **getmem** command, as shown in the following example:

```
3:ASN-9000:dec# getmem
Memory allocated for DECnet routing.
```

If memory has been allocated for DECnet routing at the time the configuration is saved with a **savecfg** command, the corresponding **getmem** command is placed in the configuration file ahead of other DECnet configuration commands. Thus, the **getmem** command need only be entered when first configuring DECnet routing.

NOTE

FORE Systems recommends that memory be allocated for the DECnet subsystem immediately after booting the ASN-9000 to ensure that the memory requested is available. For more information, refer to the *ForeRunner ASN-9000 Hardware Reference Manual.*

Memory cannot be de-allocated. To free allocated memory, make sure the configuration file does not contain a **getmem** command, then reboot the software.

Verify that memory has been allocated using the **rs** command. If memory has not been allocated, the command is not allowed to execute.

routing-status|rs [show]

```
3:ASN-9000:dec# rs
DECnet routing status:

Node/Segment            Management State   Routing State
---------               ----------------   -------------
DECnet-Forwarding       Disabled           Down
Segment  1.2            Disabled           Down
Segment  2.1            Disabled           Down
<additional rows omitted for brevity>
4:ASN-9000:dec#
```

4.1.2 Node Configuration

When placed in a DECnet internetwork, the ASN-9000 acts as a standard router, capable of connecting many different DECnet networks together. It determines the identity and location of its neighbors through standard DECnet Phase IV protocols, and finds the closest path to each. It then uses this information to route packets that arrive at the input segments.

The DECnet Phase IV routing protocol calls for each node to have an “area number” (between 1 and 63), as well as a node ID (from 1 to 1023). For Level-1 networks (consisting only of Level-1 endnodes and routers), the area numbers are identical and unused. In Level-2 networks, an “area” is defined as a collection of several nodes with identical area numbers. These areas are connected by Level-2 routers. If the ASN-9000 is configured as a Level-2 router (with the **node-type area-router set** command), it uses an extended set of routing protocols that can connect nodes from different areas. Normal nodes can only route packets directly to other nodes within their area (those with matching area numbers). If they are called upon to send a packet to a node in another area, they send it to the “nearest” Level-2 router. This Level-2 router keeps track of routes to all other Level-2 routers, as well as routes to normal nodes within its area. This two-level hierarchy allows for a larger network with manageable routing tables.

When the ASN-9000 is configured as a Level-2 router, it locates all nodes in its area *and* all other Level-2 routers. Note that this places some restrictions on the topology of the network, described below. The ASN-9000 announces itself as a Level-2 router to the normal nodes in its area so that all inter-area packets are sent through it (thus a pair of Level-2 routers are needed for inter-area packets: one in each area). If the ASN-9000 is placed into a network that uses Level-2 routing, and the ASN-9000 is to serve as a Level-2 router, be sure to turn this option on (with **`node-type area-router set`**). If the ASN-9000 is not to be a Level-2 router, or if the network uses only Level-1 routing, turn this option off (with **`node-type router set`**). The default is to use only Level-1 routing.

The use of areas in DECnet Level-2 routing places some restrictions upon the topology of these networks:

- Each node must be able to get to each other node in its area without the use of Level-2 routers and without leaving its area. Consequently, all the nodes in a given area must form a contiguous group. If all nodes from other areas are removed, leaving only the nodes from this area, there can be no isolated nodes remaining. This restriction also applies to Level-2 router nodes.
- The set of all Level-2 router nodes must form a contiguous group so that any packet going from one Level-2 router to another can travel only through other Level-2 router nodes.
- There can not be multiple links between adjacent routers. If two routers are directly connected by more than one segment, the DECnet protocol must be enabled and running on only one of those links. Failure to ensure DECnet is running on only one link results in changes to the routing table every time the doubly-connected nodes discover each other. Such a double connection causes the routing table to be continually flushed, resulting in poor performance and unreachable nodes.

This situation is represented graphically in Figure 4.1 on the next page.

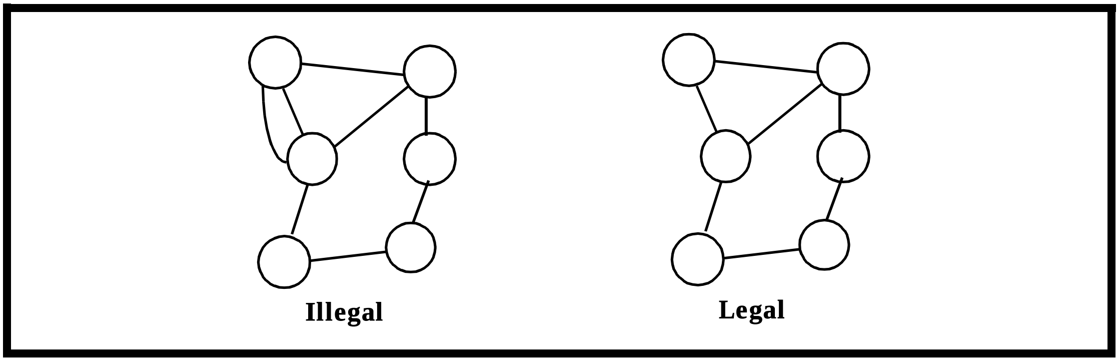


Figure 4.1 - Illegal Double Links

There is also a topological consideration that improves the efficiency of DECnet Level-2 networks. When a heavily populated broadcast medium is used, all the nodes on the same segment should be assigned the same area number. The reason is that two nodes with different area numbers must use Level-2 routing to communicate. Therefore this cable segment must have a pair of Level-2 routers on it (one for each area), and the communication path requires three hops, even though the nodes are on the same segment and could communicate directly by other protocols. To avoid these extra hops, all nodes that can communicate directly with each other should be placed in the same area by giving them identical area numbers.

4.1.3 DECnet Network Topology Restrictions

- All nodes in a given area must be connected.
- All Level-2 nodes must be connected.
- No redundant paths are allowed between adjacent routers.

Note that these restrictions do not prevent the same network segment from serving both Level-1 nodes and Level-2 nodes. Thus the same segment can serve to connect Level-1 routers, Level-1 endnodes, and Level-2 routers. The requirement is that all of an area be contiguous; nodes from different areas can be on the same segment as long as data moving within one area does not have to pass through the other area’s nodes in order to reach its destination.

4.1.4 Configuring the ASN-9000 as a DECnet Node

First, set the maximum node number used in this area. To do this, use the **`max-node-num`** parameter:

`max-node-num|mnn set` ***`<value>`***

This determines the number of nodes that can exist within the ASN-9000. The routing software ignores any packets from nodes outside this range. The default is 255, it must be increased to accommodate nodes with larger numbers. The DECnet protocol requires node numbers to be in the range 1 to 1023, the **`max-node-num`** parameter cannot be raised above 1023.

```
171:ASN-9000:dec# mnn set1023
Okay
```

Next, assign the ASN-9000 node ID. Use the **`node-id`** parameter:

`node-id|nid set <area>.<node>`

This command instructs the ASN-9000 to use the specified address for all DECnet communications. The <*area*> parameter must match the area in which the ASN-9000 has been placed; recall that the DECnet definition of "area" is the set of nodes that have the same area numbers. The <*node*> parameter can be any value that is unique among all nodes in the specified area.

```
172:ASN-9000:dec# nid set 5.1023
Okay
```

Select the type of routing that needs to be done by this router. Use the **`node-type`** parameter command:

`node-type|nt set <value>`

This command determines what kind of routing the ASN-9000 performs. If "**`router`**" is chosen, the ASN-9000 performs only Level-1 routing. A Level-1 router keeps track of nodes within its own area only, and does not try to determine routes to other areas. If it receives data for another area, it sends it to the nearest Level-2 router. The ASN-9000 acts as a Level-1 router by default.

If "**`area-router`**" is chosen, the ASN-9000 also performs Level-2 routing. Level-2 is a superset of Level-1: the node routes data to nodes within its area, as well as finds routes to other areas. All Level-2 routers find all other Level-2 routers (including those in other areas), and inter-area traffic is sent to a distant Level-2 router for local distribution.

By default, the router performs only Level-1 routing. No changes need to be made to this parameter if the ASN-9000 is going to be used as a Level-1 router. For Level-2 routing, enter: **node-type area-router set**.

```
173:ASN-9000:dec# ntset area-router
Okay
```

Activate DECnet routing with the **dec enable** command:

dec enable

This is the primary command which turns on all of the DECnet routing software. However, to have a useful configuration, specify two or more segments that use DECnet. This is accomplished with the **penable dec** <*seglist*> command, as described in.

```
191:ASN-9000:dec# penable
Okay
```

Verify the node configuration with the **node-type|nt [show]** command.

```
195:ASN-9000:dec# nt
DECnet node configuration
-------------------------
DEC-forwarding:     Enabled
Max-Area-Num:       63
Max-Node-Num:       1023
Max-Adj-Endnodes:   1023
Max-Adj-Routers:    128
Max-Cost-To-Area:   100
Max-Hops-To-Area:   16
Max-Cost-To-Node:   125
Max-Hops-To-Node:   30
Max-Visits:         60
Node-Type:          Area Rtr
Node-ID:            5.1023
Routing-State:      Up
Update-Time:        60 seconds
```

Now configure one or more segments to use DECnet forwarding.

4.1.4.1 Additional Node Commands

The following additional node commands are available to set various node parameters.

max-adj-endnodes|mae set *<value>*

Sets the number of endnode adjacencies supported by this router. The range for *<value>* is **1 - 1023**.

`max-adj-endnodes|mae [show]`

Display the results of setting the **`max-adj-endnodes|mae`** command.

max-adj-routers|mar set *<value>*

Sets the number of broadcast router adjacencies supported by this router. The range for *<value>* is **1 - 560**.

`max-adj-routers|mar [show]`

Display the results of setting the **`max-adj-routers|mar`** command.

max-area-num|man set *<value>*

Sets the maximum area number allowed in the entire network. The range for *<value>* is **1 - 63**. *<value>* must be greater than or equal to the maximum area in use.

max-area-num|man `[show]`

Display the results of setting this command, issue the following command:

max-cost-to-area|mca set *<value>*

Sets the maximum cost possible in a path to a reachable area. The range for *<value>* is **1 - 1022**. *<value>* must be greater than or equal to actual max hops to an area * **25**.

max-cost-to-area|mca [show]

Display the results of setting the **max-cost-to-area|mca set <*value*>** command.

max-cost-to-node|mcn set *<value>*

Sets the maximum cost possible in a path to a reachable node. The range for *<value>* is **1 - 1022**. *<value>* must be greater than or equal to actual max hops in area * **25**.

max-cost-to-node|mcn [show]

Display the results of setting the **max-cost-to-node|mcn set <*value*>** command.

max-hops-to-area|mha set *<value>*

Sets the maximum hops possible in a path to a reachable area. The range for *<value>* is **1 - 30**. *<value>* must be greater than or equal to actual max hops to any area.

max-hops-to-area|mha [show]

Display the results of setting the **max-hops-to-area|mha set <*value*>** command.

max-hops-to-node|mhn set *<value>*
Sets the maximum hops possible in a path to a reachable node. The range for *<value>* is **1 - 30**. *<value>* must be greater than or equal to actual max hops in an area.

max-hops-to-node|mhn [show]
Display the results of setting the **max-hops-to-node | mhn set <*value*>** command.

max-node-num|mnn set *<value>*
Sets the maximum node number allowed within this area. The range for *<value>* is **1 - 1023**. *<value>* must be greater than or equal to maximum node number in use.

max-node-num|mnn [show]
Display the results of setting the **max-node-num | mnn set <*value*>** command.

max-routers|mr pset *<value>* *<seglist>*
Sets the number of broadcast router adjacencies supported on the port(s) in *<seglist>*. *<seglist>* is a comma-separated list of ports or **all**. The range for *<value>* is **1 - 20**.

max-routers|mr [show] [*<seglist>*]
To display the results of setting this command, issue the following command:

hello-time|ht pset *<value>* *<seglist>*
Sets the interval for sending hello packets on the port(s) in *<seglist>*. *<seglist>* is a comma-separated list of ports or **all**. The range for *<value>* is **1 - 8191**

hello-time|ht [show] [*<seglist>*]
Display the results of setting the **hello-time|ht pset <*value*> <*seglist*>** command.

cost|c pset *<value>* *<seglist>*
Sets the cost for the ports in <seglist>. <seglist> is a comma-separated list of ports or all. The range for <value> is 1 - 127

cost [show] [*<seglist>*]
Display the results of setting the **cost|c pset <*value*> <*seglist*>** command.

max-visits set *<value>* *<seglist>*
Sets the maximum visits for a packet before the router assumes that the packet is looping. The range for <value> is maxpath - 60. <value> must be greater than equal to the actual maximum path in the entire network.

max-visits [show]
Display the results of setting the **max-visits set <*value*> <*seglist*>** command.

priority|pri pset ***<value> <seglist>***

Sets the priority for the port(s) in <seglist>. <seglist> is a comma-separated list of ports or all. The range for <value> is 0 - 127.

priority|pri [show] ***<seglist>***

Displays the results of setting the **priority|pre pset** command.

update-time|ut set ***<secs>***

Sets background timer for sending routing updates. The range for <secs> is 1 - 1200.

update-time|ut [show]

Displays the results of setting the **update-time|ut set** command.

4.2 Segment Configuration

Once the ASN-9000 is configured to forward DECnet packets, designate one or more segments as DECnet segments to make the software interpret and forward the correct packets. This step also causes the software to transmit and accept routing control packets over these segments, enabling it to discover neighboring endnodes and routers. There are also several parameters associated with each segment that can be set to tune network performance.

4.2.1 Configuration

From the `dec` subsystem prompt, the only necessary segment configuration step is to enable DECnet forwarding for all segments attached to DECnet networks. The **`penable dec`**<*seg-list*> command tells the software that DECnet packets may arrive over these segments and that they should be used for routing purposes:

```
dec penable<seglist>
```

This command can be used to either enable or disable DECnet forwarding for each segment. The command uses the normal <*seg-ist*> syntax, which is a hyphen- and comma- separated list of segment numbers. For example, if segments 1, 2, and 3 are to be on DECnet networks, the command is:

```
dec penable 2.1-2.4
```

```
193:ASN-9000:dec# penable 2.1-2.4
Port 2.1: Okay
Port 2.2: Okay
Port 2.3: Okay
Port 2.4: Okay
```

After enabling the segments, you can verify the segment configuration with the **`priority show`** command:

```
priority|pri [show] [<seglist>]
```

For example:

```
196:ASN-9000:dec# pri 1-2
DECnet port configuration (Port 1)
-----------------------------------
block-size:        1498
cost:              10
curr-adj-routers: 0
designated-rtr:    aa-00-04-00-1e-8a   (5.1023)
hello-time:        15 seconds
last-hello-sent:   12 seconds ago
mgmt-state:        Enabled
max-routers:       10
priority:          0
run-state:         Up
type:              Ethernet

DECnet port configuration (Port 2)
-----------------------------------
block-size:        1498
cost:              10
curr-adj-routers: 0
designated-rtr:    aa-00-04-00-1e-8a   (5.1023)
hello-time:        15 seconds
last-hello-sent:   12 seconds ago
mgmt-state:        Enabled
max-routers:       10
priority:          0
run-state:         Up
type:             Ethernet
```

At this point, verify the routing status of the DECnet software through the **`show routing-status`** command. This command shows the state of the global DEC forwarding as well as whether or not each segment is configured to route DECnet packets.

```
198:ASN-9000:dec# show rs
DECnet routing status:

Node/Port               Management State   Routing State
---------               ----------------   -------------
DEC-Forwarding          Enabled            Up

Port  1                 Enabled            Up
Port  2                 Enabled            Up
Port  3                 Enabled            Up
Port  4                 Disabled           Down
Port  5                 Disabled           Down
<remaining rows omitted for brevity>
```

In this listing, the Management State column refers to DECnet forwarding being enabled or disabled on each segment, while the Routing State column refers to the low-level hardware status. If that segment does not have a cable attached to it (and automatic segment-state detection is enabled), or if the segment has been disabled in the bridging subsystem, the Routing State shows “DOWN” instead of “UP.”

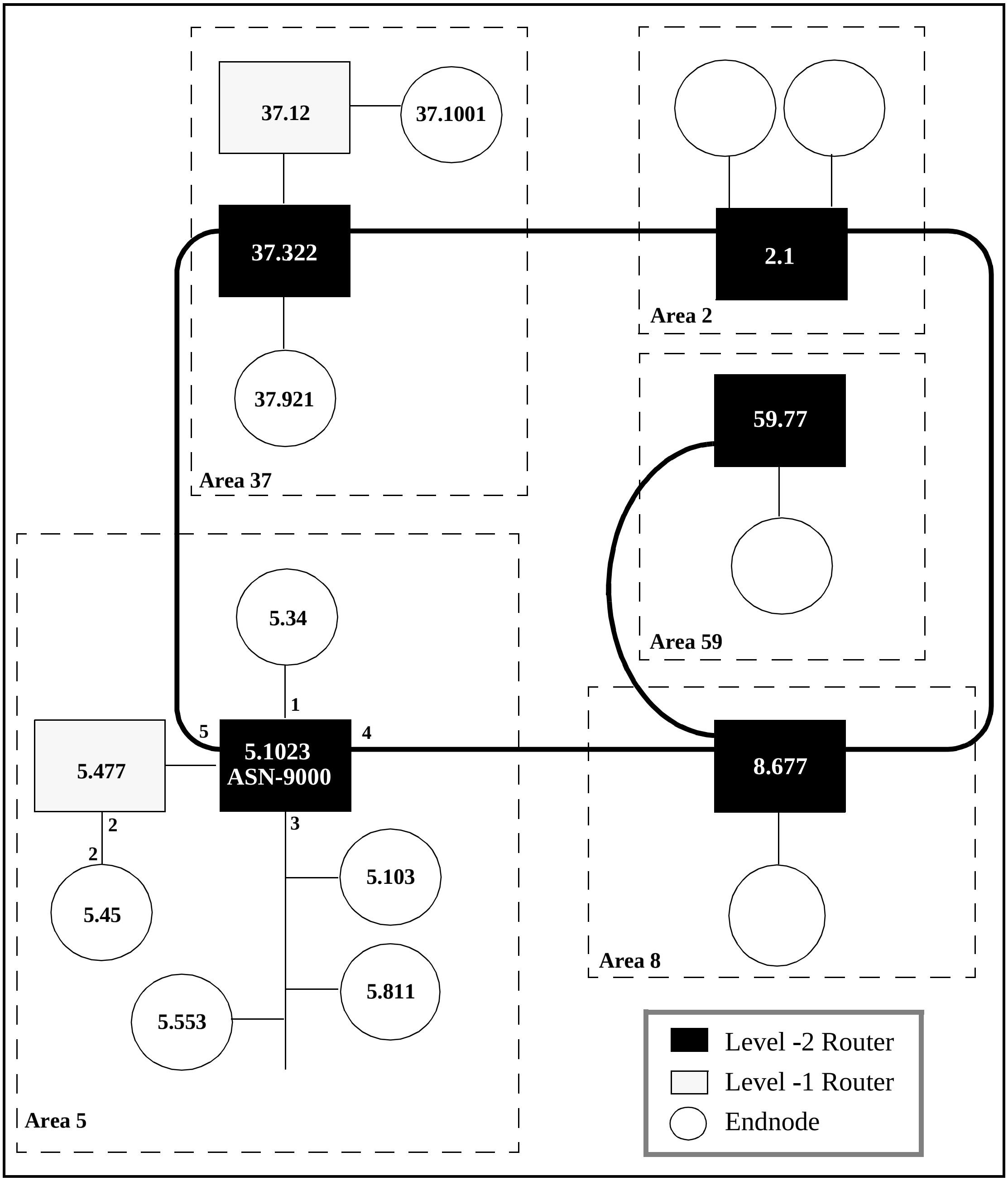
4.3 Display Commands

Using the display commands to:

- Look for adjacent DECnet routers in the network.
- Look for all DECnet endnodes adjacent to the ASN-9000.
- Look at the DECnet routing table to verify that all the routes are present.

4.3.1 Verification of Routing

After the node and segments are configured, the ASN-9000 begins forwarding packets among nearby nodes. To verify that the ASN-9000 has identified its neighbors, use one of several display commands to examine routing tables and node lists. Figure 4.2 shows a sample DECnet network. The display commands shown give information about this configuration. Note that the ASN-9000 defined as node 5.1023, located on the left, is the one being monitored. It is serving as a Level-2 router for area 5, which consists of 7 nodes: itself, 5.34, 5.477, 5.45, 5.103, 5.553, and 5.811. There are 4 other areas, 37, 2, 8, and 59. In Figure 4.2, “endnodes” are depicted as a single circle. (Endnodes, such as non-routing workstations, are nodes not capable of forwarding packets.) Level-1 router nodes are shown as lightly-shaded rectangles.


37.12
37.1001
37.322
37.921
Area 37
2.1
Area 2
59.77
Area 59
5.34
1
5
4
5.1023
ASN-9000
5.477
2
2
3
5.103
5.45
5.811
5.553
Area 5
8.677
Area 8
Level -2 Router
Level -1 Router
Endnode

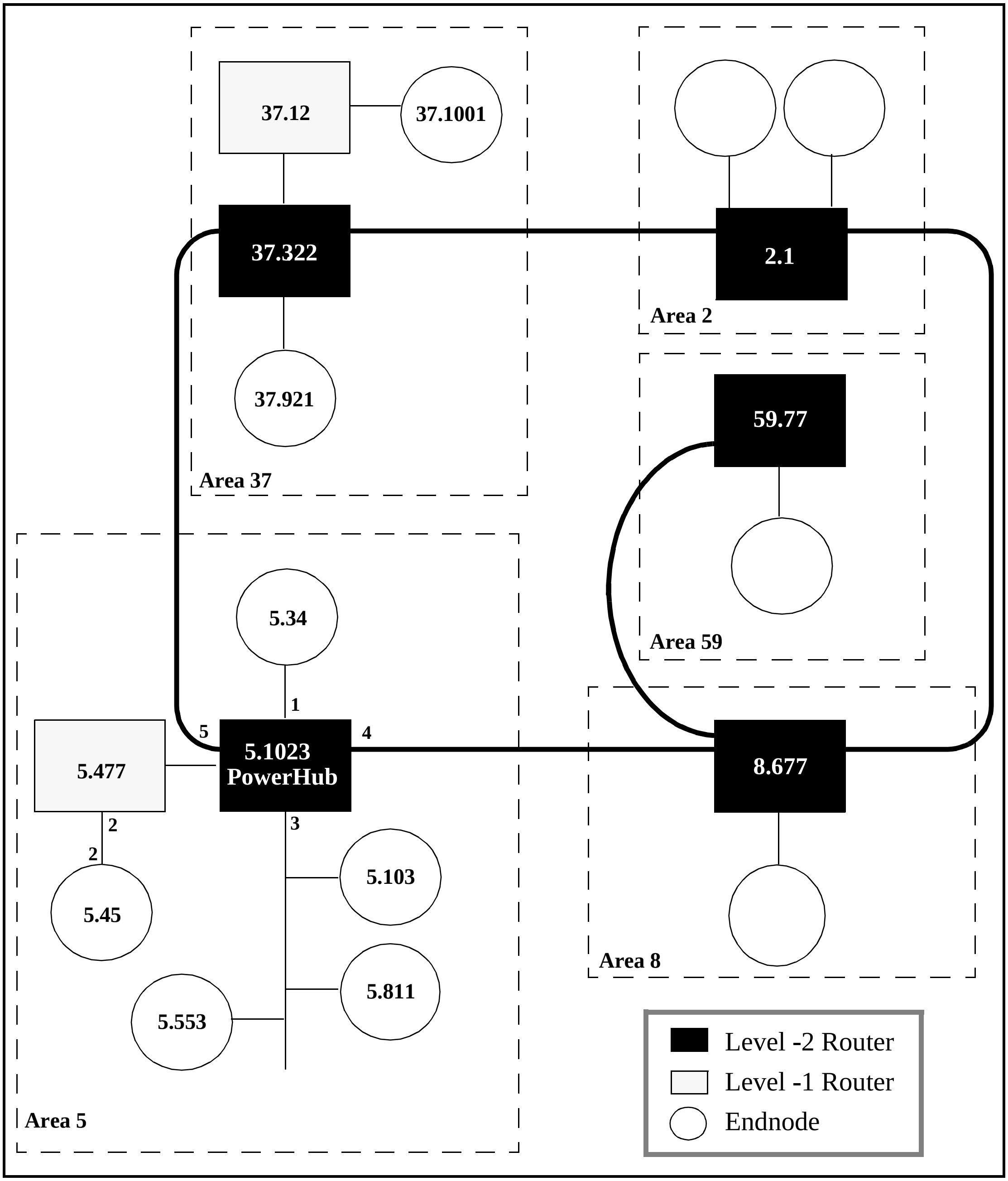


Figure 4.2 - Routing Verification

Note that nodes that are capable of routing, but appear on the periphery of networks (thus giving them nothing to route to), still qualify as routers and appear on the ASN-9000 listings as “routers” rather than “endnodes.” Level-2 router nodes are rectangles, and are connected with bold lines. All connections to the ASN-9000 are made through the segment numbers listed (1 through 5) by the small digits near the connecting lines. Also note that, while no endnodes are shown on the bold connections (links between departments, for example) between Level-2 routers, the protocols permit them to be there. For example, on the connection between 5.1023 and 37.322, endnodes or Level-1 routers for areas 5 and 37 could be attached. Each Level-2 router would recognize the nodes that belong to its area and forward packets to them.

4.3.2 Setting and Displaying Block-Size

The block-size command controls the size of internal routing tables. If it is a large network, block-size may need to be raised, but otherwise block-size should be kept low to conserve memory. To set the block-size of the internal routing tables, issue the following command:

block-size|bs pset <value> <seglist>

To display current block-size, issue the following command:

block-size|bs [show] [<seglist>]

4.3.3 Displaying Adjacent Routers

Look for adjacent routers in the network by typing:

adj[acent] [show] r[outer[s]] [[a[ddr]=]*<node>*]

```
407:ASN-9000:dec# adj r
DECnet router adjacency table:
Adj     Node ID  Type     State   Seg     Blk Siz    Hello Tim  Priority   Age
---     -------  --------------   ----   -------    -------    ------     ---
1       5.477    Router   Up      2.1     1498       15         0          3
2       37.322   Area Rtr Up      2.3     1498       15         0          3
3       8.677    Area Rtr Up      2.5     1498       15         0          3
```

This command shows all the “adjacent” routers. In DECnet terminology, “adjacent” means “directly connected.” Thus nodes on the other end of a connection are considered “adjacent.” A “router” is any node which can forward packets. Thus, this command shows all the directly connected routing nodes that the ASN-9000 has discovered. One is inside the ASN-9000 own area (5.477) and the other two are Level-2 routers (“Area Rtr”) in areas 37 and 8.

4.3.4 Displaying Adjacent Endnodes

Look for all endnodes adjacent to this router by entering:

`adj[acent] [show] [end]node[s] [[a[ddr]=]<node>]`

```
408:ASN-9000:dec# adj node
DECnet end-node adjacency table:
Adj     Node ID  Type     State   Seg  Blk Siz   Hello Tim Priority   Age
----    -------  -------- -----   ---- -------   -------  ------      ----
1       5.34     End Node Up      2.1  1498      10       0           9
2       5.811    End Node Up      2.2  1498      10       0           9
3       5.103    End Node Up      2.3  1498      10       0           9
4       5.553    End Node Up      2.4  1498      10       0           9
```

This command shows all directly connected nodes that are “endnodes,” that is, those which cannot forward packets.

4.3.5 Displaying the Route Table

Look at the routing tables to verify that all the routes

`route|rt [show] [<disprestrict>]`

This command displays the route table, which is maintained by the DECnet routing software. It contains all the routes to nodes in this area that the ASN-9000 has found dynamically (DECnet does not provide for static, user-specified routes).

Here is an example of the display produced by this command:

```
409:ASN-9000:dec# rt
DECnet routing table:
Node       Seg    Next Hop             Hops  Cost
---------  -----  -----------------    ----  ----
area-rtr   -----  This-Rtr
5.34       1.1    ------                1    10
5.45       1.2    5.477                 2    10
5.103      1.3    ------                1    10
5.477      1.2    ------                1    10
5.553      1.3    ------                1    10
5.811      1.3    ------                1    10
5.1023     Local
```

Each entry contains the following information:

Node The address of the destination node.

Port The ASN-9000 segment that a packet destined for this node should leave on.

Next Hop The address of the next node a packet must pass through.

Hops The number of nodes the packet must pass through.

Cost A number reflecting the desirability of using this route.

From this table, it can be seen that Area 5 consists of 7 nodes: 34, 45, 103, 477, 553, 811, and 1023. The ASN-9000 is node 1023. The nodes are 103, 553, and 811; they are accessible directly through segment 1.3. Two other nodes, 34 and 477, can be contacted directly through segments 1.1 and 1.2. One node, number 45, can only be reached through node 477, which is a router. Therefore the routing table shows that to send packets to node 45, the “Next Hop” is node 477, and that the node is two hops away from this one.

Note that the ASN-9000, like all DECnet nodes, keeps track of the nearest Level-2 router. Since the ASN-9000 is configured as an area-router (Level-2), the nearest Level-2 router is itself. Consequently, the next hop listed for the “area-rtr” node (the one responsible for all inter-area routing) says “This-Rtr.”

If this router is configured as an area router (Level-2), look at the area table. This is a list of all known areas, along with the best way to get to them. To display this table, issue the following command:

```
area [show] [<area>]
```

```
410:ASN-9000:dec# area
DECnet area table:
Area    Port    Next Hop     Hops      Cost
----    -----   ---------    ----      ----
2       1.4     8.677        2         20
5       Local
8       1.4     8.677        1         10
37      1.5     37.322       1         10
59      1.4     8.677        2         20
```

From this example, we can tell that the ASN-9000 is in area 5, and three other areas are accessible through the Level-2 router at 8.677, which is attached to the network on Segment 1.4. The other area is Area 37, available through segment 1.5.

If the ASN-9000 is configured as a Level-1 router (in a multi-area network), the “area-rtr” entry points to another node. As an example, imagine that the other router (node 5.477) is also a ASN-9000.

To examine the route table on that hypothetical node, something like the following is displayed:

```
1:ASN-9000:dec# area
DECnet routing table:
Nod          Segment   Next Hop     Hops    Cost
---------     -----    ---------    ----     ----
area-rtr     1.2       5.1023       1       10
5.45         1.1       5.45         1       10
5.103        1.2       5.1023       2       10
5.477        Local
5.1023       1.2       5.1023       1       10
```

Examine the node and segment statistics to verify that the ASN-9000 is receiving data and control packets correctly.

4.3.6 Displaying Statistics

There are two types of statistics collected in the DECnet subsystem: node statistics and segment statistics. The node statistics are displayed with the **stats show** command, and contain information that is not associated with any particular segment. All the numbers displayed relate to errors or dropped packets, so the ideal display is all zeros.

The syntax for the **stats show** command is as follows:

```
stats [show] <params>
<params> = p[ort] [-t] <seglist> | n[ode] [-t]
```

Here is an example of the **stats show** command:

```
411:ASN-9000:dec# stats
DECnet node statistics (count since last stats clear):
node unreachable pkt loss    0
aged packet loss             0
node out-of-range pkt loss   0
oversized pkt loss           0
pkt format error             0
partial routing update loss  0
verification reject          0
routing table corrupted      0
no timers for updates        0
no bufs for sending hello    0
invalid hello from router    0
invalid hello from endnode   0
no room for router adj       0
no room for endnode adj      0
low priority rtr bumped      0
no bufs for lvl 1 update     0
lvl 1 msg format error       0
lvl 1 msg checksum error     0
lvl 1 msg area num error     0
no bufs for lvl 2 update     0
lvl 2 msg format error       0
lvl 2 msg checksum error     0
router moved to diff. port   0
end node moved to diff. port 0
```

As in other ASN-9000 subsystems, the DECnet software maintains two copies of the node statistics:

- Count since the last clear.
- Count since the last system reset.

Both counters increment when errors occur, but the **`stats clear`** command clears only the count since last clear. To display the count since the last reset, use the **`-t`** option with the **`stats show`** command, as shown in the following example. In this particular example, the ASN-9000 has just been rebooted and no statistics have yet been collected.

```
stats [show] <params>
<params> = p[ort] [-t] <seglist> | n[ode] [-t]
```

-t Displays the count since last reset.

```
412:ASN-9000:dec# stats -t

DECnet node statistics (count since last stats reset):
node unreachable pkt loss    0
aged packet loss             0
node out-of-range pkt loss   0
oversized pkt loss           0
pkt format error             0
partial routing update loss  0
verification reject         0
routing table corrupted      0
no timers for updates        0
no bufs for sending hello    0
invalid hello from router    0
invalid hello from endnode   0
no room for router adj       0
no room for endnode adj      0
low priority rtr bumped      0
no bufs for lvl 1 update     0
lvl 1 msg format error       0
lvl 1 msg checksum error     0
lvl 1 msg area num error     0
no bufs for lvl 2 update     0
lvl 2 msg format error       0
lvl 2 msg checksum error     0
router moved to diff. port   0
endnode moved to diff. port  0
```

Segment statistics are collected in the same manner. These statistics are primarily counts of how many DECnet packets are routed through each segment. This can give an idea of where the most traffic is coming from, and may provide insight on how to better structure the network.

4.3.6.1 Displaying the Route Cache

To display the DECnet route cache, issue the following command:

```
cache [show] [<disprestrict>]
```

To clear the DECnet route cache, issue the following command:

```
cache clear
```

CHAPTER 5 Internet Protocol (IP)

This chapter describes the commands in the `ip` subsystem and shows how to use them to configure and manage the ASN-9000 as an IP router. Using `ip subsystem` commands, the following tasks can be accomplished:

- Display the IP configuration
- Add, show, and delete IP interfaces
- Enable IP routing (and allocate additional memory to the IP route table)
- Add, show, and delete IP routes
- Enable, show current settings for, and change the configuration of Router-Discovery.
- Show, add, and delete static entries from the IP Address Resolution Protocol (ARP) table.
- Ping IP workstations or other IP routers
- Add and delete IP helper addresses
- Customize the routing behavior
- Show and clear IP, ICMP, ARP, RIP, Multicast, OSPF, and IP Helper statistics
- Show or clear the IP route cache

5.1Accessing the IP Subsystem

To access the `ip` subsystem, issue the following command at the runtime command prompt:

```
ip
```

Listed below are the commands and subsystems available at this level:

```
2:ASN-9000:ip# ?
ip subsystem:

addmem                              >ospf
arp|at                              ping
arp-vlan-strict|avs                 proxy-arp
bridge-net-broadcast|bnb            rdm
cache                               >rip
config|conf                         route|rt
filter                              route-bcast-packet|rbp
fwd-pkts-with-srcrt-option|fps      route-net-broadcast|rnb
helper                              send-icmp-redirect|sir
ip|[ip] enable|[ip] disable         stats
ipdefaultttl|ittl                   template
interface|it                        traceroute
vlan                                tracesettings|tr
load-balance|lb                     tracelevel|trl
loop-detection|ld                   traceclass|trc
>mcast                              router-id|ri
```

5.2 Displaying the IP Configuration

The current IP configuration can be displayed using the **`config [show]`** command. Following is an example of the display produced by this command:

```
4:ASN-9000:ip# config show
IP Configuration:
----------------
Router ID:                         168.144.86.54
IP Forwarding:                     enabled (gateway)
Load Balancing:                    On (cache: 64, free: 64, index: 1)
Default TTL:                       64
Arp cache aging time:              5:00
Routing Network Broadcasts:        enabled
VLAN Bridging Network Broadcasts:  enabled
Routing Broadcast Packets:         disabled
Send ICMP redirects:               enabled
Forward Pkts with SrcRt Option:    enabled
Arp auto-learn:                    enabled
Arp Vlan Strict:                   disabled
Routed Packet Snooping:            disabled
```

Any of the IP configuration items listed in this display can be set.

IP Forwarding Indicates whether IP forwarding is enabled or disabled. (See Section 5.3.9.)

Load Balancing Enables the ASN-9000 to distribute IP traffic to remote destinations among up to four equal-cost routes.

When load balancing is enabled, up to four load-balancing slots are used per destination to identify next-hop gateways. Packets are hashed to a slot according to source and destination IP address so that packets belonging to a given flow always take the same path.

Default TTL Indicates the time-to-live (TTL) parameter. This parameter specifies how long a packet is allowed to remain in the net before it is dropped. (See Section 5.8.3.)

ARP cache aging time Indicates when unused learned entries in the ARP table are removed from the ARP table if they continue to be inactive. (See Section 5.6.5.)

Routing Network Broadcasts Indicates whether routing of network broadcast packets in a subnetted environment is enabled or disabled. (See Section 5.8.6.2.)

VLAN Bridging Network Broadcasts Indicates whether bridging of network broadcast packets over a VLAN is enabled or disabled.

Routing Broadcast Packets Indicates whether routing of network broadcast packets addressed to the ASN-9000 Ethernet MAC address is enabled or disabled. The default is “enabled.”

Send ICMP redirects Indicates whether Internet Control Message Protocol (ICMP) redirect messages are enabled or disabled.

Forward Pkts with SrcRt Option Indicates whether the software is permitted to forward IP packets containing source route options.

ARP auto-learn Indicates whether the software is automatically learning ARP entries. (See Section 5.6.)

5.3 Configuring and Showing IP Interfaces

Before the ASN-9000 can be used as an IP router, a vlan must be created and an IP address must be assigned to each segment through which IP packets are to be routed. When discussing TCP / IP, a connection to a physical segment is called an *interface*.

Multiple IP addresses can be assigned to the same segment. In addition, a Virtual Local Area Network (VLAN) can be created by assigning the same IP address to multiple segments. By default, IP packets are routed among different subnets, but IP packets are bridged among segments on the same subnet.

When an IP interface is configured (using the **`interface add`** command), the ASN-9000 automatically sets the MTU value for the IP interfaces.

Before configuring an IP interface, read the considerations and restrictions in Section 5.3.1 and Section 5.3.2. For information about adding IP interfaces, see Section 5.3.5.

NOTE

If theASN-9000 is configured to listen to RIP broadcasts on a subnetwork, but an IP interface address is not added to do so, a directly-attached subnet can be added.

5.3.1 Considerations

The following considerations apply to assigning interface addresses.

- An interface address must be specified in dotted-decimal notation, and it must be a valid IP host address. A valid IP address must contain a host number that is non-zero and non-broadcast (broadcast IDs are all binary 1s).
- When an IP interface is added, a subnet mask containing all ones or all zeroes can be specified.
- When an interface address is assigned to a segment, the routing software assumes that the segment is physically connected to a net whose IP network number equals the <*network-number*> part of the interface address. Routing occurs between networks with different network numbers.

NOTE

Unlike other devices, the ASN-9000 allows the same IP network number to be assigned to multiple segments (creating a VLAN). When this is done, IP packets are bridged among like-numbered nets that are connected to physically distinct segments.

- The ASN-9000 allows multiple interface addresses with different network numbers to be assigned to a single segment. When this is done, the software forwards packets for any of the corresponding nets to that segment.
- Even if routing is not desired, an interface address must be assigned to a segment in order for TELNET or SNMP connections to be made through that segment. A remote workstation uses this interface address when establishing a TELNET or SNMP connection to the ASN-9000.

5.3.2 Restrictions

The following restrictions apply when IP interface addresses are assigned. These restrictions are necessary to ensure reliable operation. Invalid configurations can bring down an entire network.

- When a single network number appears on multiple segments, all those segments must be assigned the same interface address and subnet mask.
- A parent network address cannot be configured when one or more subnets of that address have been configured on one or more segments.
- A subnet address cannot be configured if its parent network address has been configured on one or more segments. The parent network is the overall network on which subnetworks are configured. For example, network 147.128.0.0 is the parent network of subnetworks 147.128.1.0 and 147.128.2.0. These two subnetworks are referred to as children networks of the parent network.
- Different IP host addresses cannot be assigned to one interface on the same network or subnet.
- For proper operation under RIP, subnet addresses should normally all have the same binary length—in other words, they should all use the same subnet mask. If it is necessary to assign variable-length subnet addresses (different subnet masks for some addresses), certain rules must be observed. For information on these rules, see *ForeRunnerASN-9000 Filters Reference Manual.*

5.3.3 How IP Packets are Handled

- Unexpected IP broadcast packets are discarded. The IP broadcast software traps IP broadcast packets and discards them immediately if they were not expected. This feature is particularly beneficial for large networks that experience high volumes of broadcast traffic.
- IP broadcast packets that are bridged back to it are discarded. Some workstations bridge broadcast packets (including RIP packets) sent from the ASN-9000 back to the ASN-9000. The ASN-9000 IP software checks the IP source address of the incoming packet to determine whether the packet came from the ASN-9000 itself. If the packet did come from theASN-9000, the packet is discarded.
- The software routes IP broadcast packets addressed to the ASN-9000 Ethernet address. If routing of IP broadcast packets addressed to the ASN-9000 Ethernet address need to be disabled, use the **`route-net-broadcast|rnb enable|disable c`**ommand. (See Section 5.8.6.2.)

5.3.4 Configuring VLANs

In order to add an IP interface, you must first create a VLAN. A VLAN is a network that spans two or more physical segments. VLANs make network configuration changes simple by allowing the user to create and change LANs logically using software commands, as opposed to physically moving segment cables.

Any number of segments in the ASN-9000 can be defined as members of a VLAN. VLANs can overlap, so the same segments can be members of more than one VLAN. Multiple VLANs can even be defined on the same segment.

For each segment in the VLAN, the effective bandwidth available to nodes on the VLAN increases. Even though bandwidth is increased, administration and management overhead for the segments in the VLAN does not increase, because the segments can be managed as a single network.

To add a VLAN, issue the **vlan add** command. The syntax for this command is:

vlan add ***<vlanid> <seglist>***

<vlanid> Specifies the VLAN IDs to create the corresponding VLANs. Specify a single VLAN ID or a comma-separated list of VLAN IDs.

<seglist> Specifies the segments to be included in the VLAN. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

In the following example, vlan 2 has been added to segments 1.2, 1.3, and 1.4. The **vlan show** command is used to display the results of the VLAN created:

```
31:ASN-9000 :ip# vlan add 2 1.2,1.3,1.4
2:ASN-9000 :ip# vlan show
VLAN       State      Segment List
---------  ------     -----------------------
-p         down       1.10
r          down       1.9
t          down       1.7,1.8,1.9,1.10,1.11
testr      down       1.4
HR         down       1.12
techpubs    up        2.1
```

5.3.4.1 Changing VLAN Configurations

Because VLANs are created using software commands, rather than by rearranging network segments, VLANs can easily be changed to suit networking needs. Segments can be added and deleted from the configured vlans. To add segments to a VLAN, issue the **vlan tadd** command. The syntax for this command is:

vlan tadd *<vlanid> <seglist>*

<vlanid> Specifies the VLAN IDs of the corresponding VLANs to be changed. Specify a single VLAN ID or a comma-separated list of VLAN IDs.

<seglist> Specifies the individual segments to add to the VLAN. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

In the following example, segments 1.13-1.15 have been added to VLAN t:

```
5:ASN-9000:ip# vlan tadd t 1.13-1.15
vlan t modified by adding segments 1.13,1.14,1.15
```

To delete segments from a configured vlan, use the **vlan tdel** command:

vlan tdel <vlanid> <seglist>

<vlanid> Specifies the VLAN IDs of the corresponding VLANs to be changed. Specify a single VLAN ID or a comma-separated list of VLAN IDs.

<seglist> Specifies the individual segments to add to the VLAN. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

```
6:ASN-9000:ip# vlan tdel t 1.13
vlan t modified by deleting segments 1.13
```

5.3.4.2 Deleting a Configured VLAN

To delete a configured VLAN, issue the **`vlan del`** command. The syntax for this command is:

```
vlan del<vlanid>
```

<vlanid> Specifies the VLAN IDs to delete from the corresponding VLANs. Specify a single VLAN ID or a comma-separated list of VLAN IDs.

NOTE

The corresponding interface address must be deleted before the vlan can be deleted.

5.3.5 Adding an IP Interface

After creating the vlan on the segment, use the **interface add** command to assign an IP address. When an interface address is added, the software makes an entry into the IP route table to show that the corresponding network is connected to the specified segment. The software then creates the interface. The syntax for this command is:

```
it|interface add <vlanid> <ipaddr>[/<prefixlen>|<mask>]
   [ ift[ype] b[c] | n[bma] | [p[top] <nbr_addr>] ]
```

<vlanid> Specifies the VLAN ID to assign to the specified segment(s). By assigning the same IP address to multiple segments, a VLAN can be created. The IP address must be in dotted-decimal notation (four decimal numbers in the range 0–255 separated by dots).

<ipaddr> Specifies the IP address to assign to the specified segment(s). The IP address must be in dotted-decimal notation (four decimal numbers in the range 0–255 separated by dots).

<prefixlen> Allows a valid variable-length subnet to be created by using the **`interface add`** command. For more information about variable-length subnets, see Chapter 17, Configuring IP / RIP.

<mask> Allows a standard IP subnet mask to be used. If a particular network uses IP subnet addressing, then the subnet mask should be specified here using dotted-decimal notation. Otherwise, a default subnet mask equal to the “natural” subnet mask for the particular class of address is used.

[ift[ype] b[c] | n[bma]| [p[top] <nbr_addr>]]

Interface type can be one of broadcast=broadcast IP interface, nbma=Non Broadcast Multiple Access, and ptop=point to point. Neighbor address must be specified only for ptop type. If interface type is not specified, broadcast is assumed by default.

The following example below shows the use of the **`interface add`** command to add an interface:

```
14:ASN-9000:ip# it add vlan1 12.23.4.5
Vlan vlan1, Addr 12.23.4.5, Subnet mask 255.0.0.0,  type bcast Added
```

An interface address can be assigned with a non-zero cost to force routing through a desired path in the presence of redundant paths. In the following example, segments 1 and 2 are physically connected to the same router:

```
22:ASN-9000:ip# it add 1.1 147.128.132.1 255.255.255.0
Adding subnet 147.128.132.0: Okay
Port 1.1, Addr 147.128.132.1, Mask 255.255.255.0 added
23:ASN-9000:ip# it add 1.2 147.128.136.1 255.255.255.0 cost 3
Adding subnet 147.128.136.0: Okay
Port 1.2, Addr 147.128.136.1, Mask 255.255.255.0, cost 3, added
```

Because a higher cost is assigned to segment 1.2, all routing is forced through segment 1.1.

NOTE

When making changes to an IP address or subnet mask, it is not necessary to reboot the ASN-9000.

5.3.6 Deleting an IP Interface

To delete one or more interface addresses, issue the **`interface del`** command:

```
it|interface del[ete] [-p] <vlanid> <ipaddr>|all
```

[-p] Allows the address-based parameters of RIP entries to be preserved.

<vlanid> Specifies the VLAN IDs for which to delete the corresponding interfaces. Specify a single VLAN ID or a comma-separated list of VLAN IDs.

<ipaddr>|all Specifies the IP addresses for which to delete the corresponding interfaces. Specify a single address, a comma-separated list of addresses, or a hyphen-separated range of addresses. If **`all`** is specified, all IP addresses are deleted from the specified segments.

NOTE When the last interface to a particular net is deleted, that net is automatically deleted from the route table.

Following is an example of the use of this command. To delete a particular interface address on a particular segment, specify the segment number and interface address.

```
19:ASN-9000:ip# it del vlan1 12.23.4.5
Deleted interface 12.23.4.5 on vlan vlan1
```

NOTE When making changes to IP address or subnet mask, it is not necessary to reboot the ASN-9000.

5.3.7 Showing the IP Interface Table

The **`interface [show]`** command to is used display the configured IP interface addresses. For each segment, the table lists the IP addresses assigned to the segment, the link state of the segment (UP or DOWN), and other information. The syntax for this command is:

`interface|it [show][`<*disprestrictors*>**`]`**

<disprestrictors> IP address for which to display the table.

The **`it show`** command displays the following information:

```
20:ASN-9000:ip# it show
Vlan            Interface Addr  Subnet Mask     Type  Neighbor Addr   MTU   Oper
--------------- --------------- --------------- ----- --------------- ----- ----
HR              143.123.11.12   255.255.0.0     bcast --------------- 1500  down
test            146.111.111.22  255.255.0.0     bcast --------------- 1500  up
test1           147.11.22.33    255.255.0.0     bcast --------------- 1500  up
techpubs        169.144.86.54   255.255.0.0     bcast --------------- 1500  up
```

where

Vlan VLAN identifier associated with this interface.

Interface Addr IP address of this interface.

Subnet Mask Mask associated with this interface.

Type Type of IP interface. Valid types are:
bcast = broadcast IP interface
nbma = Non Broadcast Multiple Access (NBMA)
ptop = Point-to-point

Neighbor Addr IP address of neighbor (valid for ptop interfaces only).

MTU MTU for this interface. It is the minimum of the MTUs of all ports in the VLAN on which this interface is configured.

Oper Operational status of this interface.

IP Interface Count The number of IP interfaces configured.

The interface show command can also be entered with the IP address as a display restrictor. In addition, asterisks can be used as wildcards to prompt a display for a specified subnet only. The command below show the use of the wildcard option:

```
4:ASN-9000:ip# it show addr=169.*.*.*
Vlan           Interface Addr Subnet Mask    Type    Neighbor Addr  MTU     Oper
--------------------------------------------------   --------------------   ----
techpubs       169.144.86.54  255.255.0.0    bcast    ---------------1500   up
```

5.3.8 Allocating Memory for Additional IP Routes

Before the `ip` subsystem commands can be used, memory must be allocated for the subsystem by issuing the **`addmem`** command. Memory allocation increases the capacity of the IP route table. Additional memory can be specified in terms of IP routes. The increment is 1K routes. The following example shows the results of this command:

```
20:ASN-9000:ip# addmem
IP: successfully allocated memory for 1024 additional routes
```

If memory has been allocated for IP routing at the time the configuration is saved with a **`system savecfg`** command, the corresponding `ip` subsystem **`addmem`** command is placed in the configuration file ahead of the other `ip` commands. Thus, it is only necessary to type the **`addmem`** command when the ASN-9000 is first configured for `ip` routing.

5.3.9 Enabling IP Routing

Since IP routing is by default disabled, IP routing can be enabled after defining the IP interfaces (see Section 5.3), using the following command:

```
ip enable
```

5.4 Showing, Adding, and Deleting IP Routes

This section describes how to display the IP route table and interpret its contents. This section also describes how to manually add and delete static route-table entries. Note that the software makes additions to the IP route table in two basic ways: it “learns” them from a routing protocol (RIP or OSPF) or they are added manually. Learned routes are called “dynamic entries” and user-added routes are called “static entries.”

5.4.1 Showing the IP Route Table

To display the IP route table, issue the following command:

```
route|rt [show] [-c|-r|-s|-i|-o] [-d|-t] [-a] [<disprestrictors>]
```

[-c|-r|-s|-o] Filters the display according to the type of route:

-cDisplays only directly connected entries.

-rDisplays only RIP routes.

-sDisplays only static routes.

-iDisplay special routes

-oDisplays only OSPF routes.

[-d|-t] Displays additional information, including statistics for packets and bytes. When this argument is specified, the **`-f`** argument is ignored. -**`t`** displays the total number of routes.

[-a] Displays only active routes.

<disprestrictors> addr[ess]=<ipaddrlist>

nh[op]=<ipaddrlist>

The **rt show** command will display the following information:

```
2:ASN-9000:ip# rt
Destination      Subnet Mask    Gateway        Met   Prf    State   RtSrc   Flg     Age
---------------  ----------------------------------- ---    ------  ------- ---     ---
146.111.0.0      255.255.0.0    146.111.111.22 0     0      Active  direct  LOC
146.111.111.22   255.255.255.255146.111.111.22 0     0      Active  direct  MYA
147.11.0.0       255.255.0.0    147.11.22.33   0     0      Active  direct  LOC
147.11.22.33     255.255.255.255147.11.22.33   0     0      Active  direct  MYA
147.111.0.0      255.255.0.0    147.111.111.22 0     0      Active  direct  LOC
147.111.111.22   255.255.255.255147.111.111.22 0     0      Active  direct  MYA
147.123.0.0      255.255.0.0    147.123.22.34  0     0      Active  direct  LOC
147.123.22.34    255.255.255.255147.123.22.34  0     0      Active  direct  MYA
169.144.0.0      255.255.0.0    169.144.86.54  0     0      Active  direct  LOC
169.144.86.54    255.255.255.255169.144.86.54  0     0      Active  direct  MYA
Total routes: 10 (Direct: 10, Static: 0, Special: 0, RIP: 0, OSPF: 0)
Active Routes: 10
Backup Routes: 0
Down Routes: 0
Free Routes: 4085
```

Destination IP address of the destination host or network.

Subnet Mask Subnet mask of destination host or network.

Gateway Nexthop to be used for forwarding to destination.

Met Metric for this route (protocol specific).

Prf Internal preference for this route. The lower the value the better the route, i.e., routes with a lower preference are more likely to become the active route and hence used for forwarding.

State Indicates whether this route is usable or not, where:
Active = route available for forwarding traffic
Backup = route not being used but could be used
Down = route is disabled or cannot be used

RtSrc Source of the routing information.

Flg Tells how packets to this destination are handled.

Age Age of the route (valid for RIP only).

Active Routes Number of active routes.

Backup Routes Number of backup routes.

Down Routes Number of down routes.

Free Routes Number of routes that can yet be added to the routing table.

The **route show** command can also be entered with asterisks to limit the number of interfaces displayed to specified subnets. The example below shows this command:

```
10:ASN-9000:ip# rt show addr=169.*.*.*
Destination      Subnet Mask     Gateway     Met    Prf     State  RtSrc  Flg    Age
---------------  --------------- -----------------------    ------ --------------    -
169.144.0.0      255.255.0.0     169.144.86.540     0       Activ  direct LOC
169.144.86.54    255.255.255.255 169.144.86.540     0       Active direct MYA

Total routes: 2 (Direct: 2, Static: 0, Special: 0, RIP: 0, OSPF: 0)
Active Routes: 2
Backup Routes: 0
Down Routes: 0
Free Routes: 4085
Allocated Routes: 4096
```

5.4.2 Adding and Deleting IP Routes

The ASN-9000 stores information about routes in the route table. Entries in the route table are either learned dynamically by RIP (as described in Section 5.12), or entries can be configured into the table manually (static entries). Static routes can be assigned for individual hosts or for entire nets.

All nets that have corresponding interface addresses assigned to one or more ASN-9000 segments are considered to be directly attached.When such interface addresses are assigned by the **interface add** command, a corresponding entry is automatically made in the route table. As a result, the routing software automatically routes any incoming IP packet whose destination address is on a directly attached net to the corresponding segment(s). No additional configuration is required.

Additional information is required, however, to route packets to destinations that are not directly attached. In many cases, routers can use RIP to dynamically discover routes that are not directly attached to the hosts and nets. Routes also can be statically assigned, as described in this section. If RIP is not running, routes to non-directly-attached hosts and nets must be assigned statically. To assign the route to be used when forwarding to a host or net, use the **route add** command. To delete a route, use the **route del** command:

```
route|rt add <destination> <gateway> [metric <metric>] [pref <pref>]
            route|rt del[ete] <destination> <gateway>
```

where

<destination> Can be one of:
<ipaddr>/<prefixlength> (Ex: 10.0.0.0/8)
<ipaddr>/<mask> (Ex: 10.0.0.0/255.0.0.0)
host <ipaddr> (Ex: <ipaddr>/32)
default

<gateway> IP address of gateway (next hop) for this route.

<metric> Route metric (number of hops to the destination).

<pref> Internal cost for this route. The lower the value, the better the route, i.e., routes with a lower preference are more likely to become the active route and hence used for forwarding.

The example below shows how this command:

```
18:ASN-9000:ip# route add 100.1.1.0/24 10.1.1.2 metric 2
```

5.4.3 Enabling and Disabling Load Balancing

When load balancing is enabled, the ASN-9000, receiving packets from the same source, uses different routes for the incoming packets to reach the ASN-9000 without any delay. To enable load balancing, issue the following command:

```
load-balance|lb enable|disable
```

enable|disable Specifies whether to enable or disable load balancing. The default is disable.

The examaple below shows how this command is entered:

```
28:ASN-9000:ip# lb enable
IPR: load balancing is turned on
```

5.4.4 Enabling Loopback Detection

When loopback detection is enabled, the ASN-9000 sends a special loopback-detect packet on each outbound segment that has at least one IP address. To enable loop detection on theASN-9000, issue the following command:

`loop-detection|ld enable|disable`

enable|disable Specifies whether to enable or disable loop detection.

Following is an example of this command:

```
69:ASN-9000:ip# ld enable
loop-detection:         enabled
```

5.4.4.1 Setting the Loopback Detection Time

The **`loop-detection set time`** command is used to set the time interval for sending out loopback-detection packets. To set the loopback detection time, issue the following command:

`loop-detection|ld set time <value>`

<value> Specifies the time interval in minutes for sending out loopback-detection packets. The default is 10 minutes.

Following is an example of this command:

```
70:ASN-9000b:ip# ld set time 15
71:ASN-9000:ip#
```

5.4.4.2 Enabling and Disabling Loopback Detection

Enable or disable the loopback-detection time by using the following syntax:

`loop-detection|ld enable|disable`

The example below shows this command:

```
31:ASN-9000:ip# ld enable
loop-detection has now been enabled
```

5.4.4.3 Displaying the IP Loop Detection Table

To display the IP loop detection table, issue the following command:

`loop-detection|ld [show]`

Following is an example of this command:

```
72:ASN-9000:ip# ld show
loop-detection: enabled
IP Loop Detection Table:
IP Address         MAC Address          TTL     rport       Segment(s)
---------------    -------------------  ---     -----       -------------
147.128.128.2      08-00-20-08-70-54    16      2           1.3
```

For each IP route, the route table shows the following information:

IP Address The IP address of the outbound segment sending the loopback-detect packet.

MAC Address The Ethernet address of the host.

TTL Specifies how long a packet is allowed to remain in the net before it is dropped. Packets that cannot find or are blocked from their destination nodes are dropped when the TTL expires.

rport Specifies the receiving port of the segments sending the loopback-detect packet.

Segment (s) The segment(s) that sent the loopback-detect packet.

5.4.5 Enabling or Disabling an IP Route

After IP interfaces (see Section 5.3) have been assigned, static routes can be enabled or disabled using the following command syntax:

```
route|rt enable <destination> <gateway>
route|rt disable <destination> <gateway>
```

enable|disable Specifies whether to enable or disable static route. The default is disable.

<destination> Can be one of:
<ipaddr>/<prefixlength> (Ex:10.0.0.0/8)
<ipaddr>/<mask>(Ex: 10.0.0.0255.0.0.0)
host<ipaddr>(Ex:<ipaddr>/32)
default

<gateway> Specifies the IP address of the gateway (next-hop router) to which packets destined for the specified host are forwarded. Generally, this gateway is connected to the ASN-9000 through a net. The net is directly attached to both the gateway and the ASN-9000.

5.5 IP Router Discovery

Based on the ability of ICMP (internet control message protocol) to enable hosts attached to multicast or broadcast networks to discover the IP addresses of their neighboring routers, router discovery allows hosts to discover routers automatically through a series of solicitation and advertisement messages. This eliminates the need for the specific configuration of static addresses.

Before a host can send IP datagrams beyond its directly-attached subnet, it must discover the address of at least one operational router on that subnet. Typically, this is accomplished by reading a list of one or more router addresses from a (possibly remote) configuration file at start-up time. On multicast links, some hosts also discover router addresses by listening to routing protocol traffic. Routing discovery on the ASN-9000 uses a pair of ICMP [10] messages for use on multicast links. More information about router discovery can be found in RFC 1256. To enable or disable router discovery on the ASN-9000, issue the following command:

```
rdm nenable|ndisable <ipaddr>
```

<ipaddr> Specifies the IP address of the host.

The example below enables routing discovery on the specified interface:

```
38:ASN-9000:ip# rdm nenable 146.111.111.22
```

5.5.1 Setting the Advertisement Address

Whether to send out advertise messages to the all-systems multicast address, 224.0.0.1, or to the limited-broadcast address, 255.255.255.255, can be specified. By default, the ASN-9000 designates the all-systems multicast address. To set the Advertisement Address for Router Discovery on the ASN-9000, issue the following command:

```
rdm nset AdvertisementAddress  multicast|broadcast <ipaddr>
```

AdvertisementAddress Specifies the IP destination address to be used for multicast Router Advertisements sent from the interface. The only permissible values are the all-systems multicast address, 224.0.0.1, or the limited-broadcast address, 255.255.255.255.

multicast|bro Specifies advertise messages to all-systems multicast address, 224.0.0.1, or to the limited-broadcast address, 255.255.255.255. The ASN-9000 default is all-systems multicast, 224.0.0.1.

<ipaddr> Specifies the IP address belonging to the interface from which this message is sent, or 0.

5.5.2 Setting the Advertisement Preference

A Router Advertisement includes a preference level for each advertised router address. When a host must choose a default router address for a particular destination and the host has not been redirected or configured to use a specific router address, the host is expected to choose from those router addresses that have the highest preference level. To set the advertisement preference for Router Discovery on the ASN-9000, issue the following command:

```
rdm nset preference <preference> <ipaddr>
```

preference Specifies the preference value for Router Discovery.

<preference> Specifies the preference of each Router Address as a default router address, relative to other router addresses on the same subnet. A signed, twos-complement value; higher values mean more preferable.

A 32-bit, signed, twos-complement integer, with higher values meaning more preferable. The minimum value (hex 80000000) is used to indicate that the IP address, even though it may be advertised, is not to be used by neighboring hosts as a default router address.

<ipaddr> Specifies the IP address belonging to the interface from which this message is sent, or 0.

5.5.3 Setting the Advertisement Interval

A Router Advertisement also includes a lifetime field, specifying the maximum length of time that the advertised addresses are to be considered as valid router addresses by hosts in the absence of further advertisements. This is used to ensure that hosts eventually forget about routers that fail, become unreachable, or stop acting as routers. The default advertising rate is once every 10 minutes, and the default lifetime is 30 minutes. To set the advertisement interval for Router Discovery on the ASN-9000, issue the following command:

```
rdm nset interval <time> <ipaddr>
```

interval Specifies the interval value for Router Discovery.

<time> Specifies the time allowed between sending multicast Router Advertisements from the interface.

<ipaddr> Specifies the IP address belonging to the interface from which this message is sent, or 0.

5.5.4 Displaying the Advertisement Interval

To display the Router Discovery table, issue the following command:

```
rdm [show]
```

Here is an example of the results produced by this command:

```
72:ASN-9000:ip# rdm show
 -- RDM Configuration --
Interface      Dest Address  Interval  Preference   Advertise
---------------  ------------  --------  -----------  ---------
143.123.11.12  multicast      10:00          0        yes
146.111.111.22 multicast      10:00          0        yes
147.11.22.33   multicast      10:00          0        yes
147.111.111.22 multicast      10:00          0        yes
147.123.22.34  multicast      10:00          0        yes
169.144.86.54  multicast      10:00          0        no
147.128.124.4  multicast      10:00          0        yes

IP Interface Count: 6
```

5.6 Showing and Configuring the ARP Table

The ASN-9000 IP routing software maintains an ARP table of IP-to-Ethernet address translations. These translations are used to route packets and, under some circumstances, to generate replies to ARP requests.There are three ways that entries are added to the ARP table:

- When a host uses ARP to request the ASN-9000 Ethernet address, the host’s IP and Ethernet addresses are recorded (“learned”).
- If a host forwards a packet to a destination through the ASN-9000, it can generate an ARP request to learn the destination’s Ethernet address. When a reply to such a request is received, it records the destination’s IP and Ethernet addresses.
- Permanent entries are added using ASN-9000 commands.

5.6.1 Enabling and Disabling ARP

To enable ARP auto-learning, issue the following command:

```
arp enable auto-learn
```

auto-learn Indicates enabling auto-learn of incoming packets on the ASN-9000. Default is auto-learn enabled.

The command below enables auto-learning:

```
57:ASN-9000:ip# arp enable auto-learn
```

To disable ARP auto-learning, issue the following command:

```
arp disable auto-learn
```

The command below disables auto-learning:

```
58:ASN-9000:ip# arp disable auto-learn
```

5.6.2 The ARP Cache

IP route packets are queued for which the ARP table does not contain entries, then sends an ARP request to learn the Ethernet address of the destination device. When the ARP reply is received from the destination device, the queued packet is forwarded. The source node does not need to resend the packet.

5.6.3 Showing the ARP Table

The **`arp [show]`** command is used to display the current contents of the ARP table. The syntax for this command is:

`arp [show] [-r] [-t] [-s] [`*`<disp-restrictors>`*`]`

[-r] Specifies raw entries with hash indices and displacements.

[-t] Specifies that only the total count of entries is to be displayed.

[-s] Specifies that the ARP entries to be displayed are sorted by the IP address (in increasing order).

[<disp-restrictors>] address=<IPaddrlist> Specifies for which IP addresses to display the ARP table.

[[seg[ment[s]]]=]<seglist>Specifies the segments for which to display the ARP table. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

Following are some examples of the use of this command. If no argument is given, then the entire table is displayed, for example:

```
70:ASN-9000:ip# arp show
IP Addr        Ethernet Address      Flags           Segment
147.128.128.2  08-00-20-08-70-54     perm publish    6
147.128.128.3  08-00-20-08-85-69                     2
192.9.201.1    02-cf-1f-90-40-23                     12
192.9.201.7    08-00-20-0f-dd-99     perm            10
```

Permanent entries can have a flag indicating that the entry was added automatically, and a broadcast flag, indicating that the Ethernet address is broadcast or multicast.

An optional IP address can be specified with the **`arp [show]`** command, in which case only the entry for that address is displayed:

```
70:ASN-9000:ip# arp show address=147.128.128.2

IP Addr          Ethernet Address      Flags      Segment
147.128.128.2    08-00-20-08-70-54                  6
```

A “wildcard” character (**`*`**) can be used in place of any byte(s) of the IP address, in which case only entries that match that address are displayed.

Table entries for a specific segment can be viewed by using the segment option in the command:

```
68:ASN-9000:ip# arp 1.1
IP Address         Ethernet Address   Flags                  Segments
---------------    -----------------  --------------------   -------------------
147.111.111.22     00:00:ef:03:9a:b0  perm publish system    1.1
147.11.22.33       00:00:ef:03:9a:b0  perm publish system    1.1
147.123.22.34      00:00:ef:03:9a:b0  perm publish system    1.1
146.111.111.22     00:00:ef:03:9a:b0  perm publish system    1. 1

Total ARP entries: 4, perm entries: 4, learned entries 0
```

5.6.4 Clearing the ARP Table

The **arp clear** command is used to clear the ARP table. All learned entries are removed, but static entries (created using the **arp add** command) remain in the table. These must be removed manually using the **arp del** command.

This command can be used to help restabilize the network after a host is moved from one segment to another. When there is activity on the network, the cleared entries quickly reappear in the ARP table, and a host that has been moved is relearned on its new segment.

5.6.5 Showing and Changing the ARP Aging Interval

By default, the ASN-9000 automatically checks learned entries in the ARP table every five minutes to see if they have been used. Each unused entry is marked aged. If an aged entry is used during the next five-minute interval, the aged flag is removed. However, aged entries that remain unused during the second five-minute interval are removed from the ARP table.

The aging interval can be changed or turned off using the **arp set age** command. The syntax for this command is:

arp set|nset age *<time>*

<time> Specifies (in minutes) a new aging interval or turns aging off. The default is 5 minutes. Set the aging time at a minimum of 1 minute (enter either 60 (seconds) or 1:00). To specify minutes, specify <minutes:seconds>.

NOTE If ARP aging is turned off, the ARP table can quickly overflow. Make sure to monitor the table frequently if ARP aging is turned off.

Following is an example of the use of this command:

```
73:ASN-9000:ip# arp set age 30:00
ARP cache aging set to 30 minutes
```

To display the current ARP aging interval, issue the **config show** command.

```
72:ASN-9000:ip# config
IP Configuration:
----------------
Router ID:                         168.144.86.54
IP Forwarding:                     enabled (gateway)
Load Balancing:                    On (cache: 64, free: 64, index: 1)
Default TTL:                       64
Arp cache aging time:              30:00
Routing Network Broadcasts:        enabled
VLAN Bridging Network Broadcasts:  enabled
Routing Broadcast Packets:         disabled
Send ICMP redirects:               enabled
Forward Pkts with SrcRt Option:    enabled
Arp auto-learn:                    enabled
Arp Vlan Strict:                   disabled
Routed Packet Snooping:            disabled
```

5.6.6 Adding a Static Entry to the ARP Table

The **arp add** command is used to add a static ARP entry to the ARP table. Static ARP entries are not subject to aging and are not cleared when the ARP table is cleared (i.e. by using the **arp clear** command). The syntax for this command is:

arp add [-p] ***<ipaddr> <ethaddr> <seglist>***

[-p] If this argument is present, then the IP routing software replies directly to ARP requests for this entry. Note that this facility is provided only for permanent, not learned, entries in the ARP table.

<ipaddr> Specifies the IP address to be translated.

<ethaddr> Specifies the Ethernet address corresponding to the given IP address.

<seglist> Specifies the segments to which packets sent to the IP address specified by <*ethaddr*> are forwarded. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments. If **all** is specified, the ARP entry is added to all segments.

The example below shows the **arp add** command:

```
18:ASN-9000:ip# arp add 147.128.23.2 08-00-02-03-04-05 1.8
Added/Changed:
    IP Address       : 147.128.23.2
    Ethernet Address : 08:00:02:03:04:05
    Flags            : PERMANENT PUBLISH DOWN
    Segments         : 1.8
```

5.6.7 Deleting a Static Entry from the ARP Table

The **arp delete** command is used to delete static or dynamically learned ARP entries from the ARP table. The syntax for this command is:

arp delete *<ipaddr>*

<ipaddr> Specifies the IP address in the ARP entry to delete.

When a host is moved from one segment to another, this command can be used to delete its obsolete learned entry from the ARP table without disturbing any other entries. The network can then relearn the host’s new location without being forced to relearn other host locations, as it would if the ARP table was cleared.

The example below shows the **arp del** command:

```
79:ASN-9000:ip# arp del 147.128.23.2
ARP entry for 147.128.23.2 deleted
```

5.7 Pinging Other IP Devices

The ASN-9000 supports the echo facility of the ICMP in two ways:

- A response is generated to any ICMP echo request packet received on any segment.
- An ICMP echo request packet can be sent to any IP address.

ICMP echo requests are commonly used to determine whether devices are reachable on the network. UNIX workstations provide a **ping** command that generates an ICMP echo request to a specified IP address.

When the **ping** command is issued from a workstation, the ASN-9000 responds and can determine whether the ASN-9000 is reachable from the workstation. However, depending upon the configuration, the ASN-9000 might be known by multiple IP addresses. Unless the workstation is directly connected, the IP address specified in the **ping** command can affect the route taken and, therefore, the reachability of the ASN-9000.

Similarly, the ASN-9000 itself provides a ping command to generate an ICMP echo request to a specified IP address. The syntax for this command is:

ping|[-t *<timeout>*] [-size *<size>*] *<ipaddr>*

[-t <timeout>] Specifies how many seconds the ASN-9000 waits for a response from the specified device. The default is **5** seconds.

[-size <size>] Specifies the packet length. Specify any length from **64** through **1472** bytes. The default is **64** bytes.

<ipaddr> Specifies the IP address of the distant device.

Here are some examples of the use of this command.

```
83:ASN-9000:ip# ping 147.128.128.8
147.128.128.8 is alive
84:ASN-9000:ip# ping 147.128.128.15
No response from 147.128.128.15
```

The **ping** command normally waits 5 seconds for the specified host to respond before timing out. However, a shorter or longer time-out can be specified, as shown in the following example. In this example, a one-second delay is specified.

```
85:ASN-9000:ip# ping 147.128.128.8 1
No response from 147.128.128.8
```

5.8 IP Helper

This section describes how to use the IP Helper feature. IP Helper is an enhancement to the IP subsystem that assists client stations on one network segment in communicating with servers on another network segment when the two segments are connected by a ASN-9000. This includes situations where one switch, as a client station, needs to boot from a server from which it is separated by another switch.

By default, the IP Helper feature is configured to help packets destined for any of the following standard UDP ports:

- BootP client packets (port 68).
- BootP server packets (port 67).
- Domain Name System (port 53).
- IEN-116 Name Server (port 42).
- NetBIOS Datagram Server (port 138).
- NetBIOS Name Server (port 137).
- TACACS service (port 98).
- TFTP (port 69).
- Time service (port 37).

If it is necessary to add a UDP port to this list, use the **`helper add -d`** command. (See Section 5.8.2.)

5.8.1 How IP Helper Works

When a client sends a broadcast packet addressed to a server that is directly connected to the client, the server does the following:

- Receives the limited broadcast IP packet sent out by the client.
- Uses the client's Ethernet address to look up its corresponding IP address.
- Sends a unicast packet in reply.

This is also true if the client and server are on different segments but the segments are defined as part of the same VLAN. In this case, the packets are bridged.

However, if the client and server are on different segments separated by a router (gateway), the client’s broadcast packet never reaches the server. If the intervening router is an ASN-9000, the IP Helper facility on that ASN-9000 can be used to tell it where to forward UDP packets sent by the client.

To use IP Helper to help a client reach its server, assign the server's IP address as an IP Helper address to the ASN-9000 segment connected to the client. When this segment receives a UDP packet from the client, it forwards the packet to the node that has the IP address corresponding to the ASN-9000 segment's IP Helper address.

For the UDP packet to be successfully forwarded, the following criteria must be met:

- The packet must be received on a segment where an IP Helper address is configured.
- The destination UDP port must be in the UDP-helper Port Table on the router. See RFC 1542 for more information.

Because Boot packets (used for netbooting) are UDP packets, IP Helper makes netbooting possible when the client switch and server are separated by a router. Similarly, it facilitates netbooting of diskless workstations.

NOTE

IP Helper does not affect the forwarding of limited-broadcast packets in a virtual LAN environment. The same packet can be forwarded to multiple segments that are on the same virtual LAN.

5.8.2 Using IP Helper

Before you can use IP Helper:

- The ASN-9000 switch must be configured as an IP router. (See Section 5.3.)
- An IP Helper address must be assigned to the segment that connects to the diskless workstation or other device that is being helped. The IP Helper address is the address of the desired server on the network.

To display helper configuration on an IP segment, issue the following command:

```
conf [show] helper
```

5.8.2.1 Adding an IP Helper Address

To add an IP Helper address to a segment, issue the following command:

```
helper add <IPaddr> [<UDPportlist>] <seglist>
      helper add -d <more UDP ports>
```

<IPaddr> Specifies the helper address. Specify the IP address of the server as the helper address.

[<UDPportlist>] Specifies any of the standard UDP ports available by default.

<seglist> Specifies the segments on which to add an IP address. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments. If **all** is specified, then the IP address is assigned to all valid segments.

-d <more UDP ports> Displays the contents of the default UDP portlist. Allows you to specify additional default UDP ports.

Following is an example of the use of this command.

```
11:ASN-9000:ip# helper add 147.128.42.37 1.4
Helper address 147.128.42.37: added.
```

An IP Helper address is added to segment 1.4 on the router. When used, this IP Helper address routes UDP packets received on segment 1.4 to IP address 147.128.42.37.

Multiple IP Helper addresses can be assigned to a single segment, or multiple segments to a single IP Helper address. Assigning multiple IP Helper addresses to a single segment provides redundancy when multiple servers are used.

5.8.2.2 Deleting an IP Helper Address

To delete an IP Helper address, issue the **`helper delete`** command. The syntax for this command is:

```
helper delete <IPaddr> <UDPportlist>|default[s]|all <seglist>
```

<IPaddr> Specifies the helper address. Specify the IP address of the server as the helper address.

<UDPportlist>|default[s]|all Specifies the type of UDP port being deleted. If **`all`** is specified, all UDP ports, including the default ports are deleted.

<seglist> Specifies the segment(s) that connect the router to the client ASN-9000. If **`all`** is specified, all entries assigned to the specified IP address are deleted.

Following is an example of the use of this command:

```
16:ASN-9000:ip# helper delete 2.2.2.2 1.2
2.2.2.2:138 (netbios-dgm), port 1.2 :deleted
2.2.2.2:138 (netbios-ns), port 1.2 :deleted
2.2.2.2:138 (tacnews), port 1.2 :deleted
2.2.2.2:138 (tftp), port 1.2 :deleted
2.2.2.2:138 (dns), port 1.2 :deleted
2.2.2.2:138 (name), port 1.2 :deleted
2.2.2.2:138 (time), port 1.2 :deleted
```

5.8.2.3 Displaying Statistics and the UDP Table

To display current statistics for an IP Helper address defined for a segment, issue the `helper show` command. A table is displayed listing the segment, helper address, the number of packets helped, and the number of packets dropped. The syntax for this command is:

```
helper show   [-p|-s]
  helper show   -d
```

[-p|-s] Sorts the IP Helper table by UDP port **`-p`**, or by segment number **`-s`**.

-d Displays the contents of the default UDP portlist. Allows additional default UDP ports to be specified.

Following is an example of the use of this command:

```
11:ASN-9000:ip# helper show
Helper IP      UDP portSegment   Helped   Reverse  Dropped
-------------  -------- --------  -------  -------  ---------
147.128.48.37 37 time 1.4        4  0      1
```

The table in this example shows that during the current session, IP Helper address 147.128.48.37 has helped four UDP packets (perhaps BOOTP packets) find their IP destinations. The table also shows that one UDP packet was dropped. Note that the **helper show** command lists statistics only for those UDP packets that the ASN-9000 tried to help. UDP packets can be dropped for any of the following reasons:

- The helping ASN-9000 does not have a route to the destination address in the UDP packet.
- The helping ASN-9000 runs out of resources to redirect the packet.

In addition, for BOOTP packets only, the following conditions can cause the helping ASN-9000 to drop the packet:

- The hop count in the packet has been exceeded.
- A gateway has already helped the packet. (A bit in the packet is set when the packet is helped.)

The **helper -p** command sorts the IP Helper table by UDP port:

```
9:ASN-9000:ip# helper -p
Helper IP      UDP port          Segment     Helped   Reverse      Dropped
--------------------------------- -------     ------   -------      --------
2.2.2.2        37 time           1.5         0        0            0
2.2.2.2        42 name           1.5         0        0            0
2.2.2.2        53 dns            1.5         0        0            0
2.2.2.2        67 bootps         1.5         0        0            0
2.2.2.2        68 bootpc         1.5         0        0            0
2.2.2.2        69 tftp           1.5         0        0            0
2.2.2.2        98 tacnews        1.5         0        0            0
2.2.2.2        137 netbios-ns    1.5         0        0            0
2.2.2.2        138 netbios-dgm   1.5         0        0            0
```

The **Helper -d** command displays the default UDP helper portlist:

```
10:ASN-9000:ip# helper -d
Default UDP helper ports:

    37        42        53        67        68
    69        98        137       138
```

5.8.2.4 Deleting Default UDP Entries

To delete default UDP entries, issue the **`helper delete`** command. The syntax for this command is:

`helper delete -d` ***`<UDP ports to remove>`***

-d Displays the contents of the default UDP portlist. Allows additional default UDP ports to be deleted.

The **`helper delete -d`** command below removes default UDP port entry 37. The **`Helper -d`** command displays the change in default UDP entries:

```
12:ASN-9000:ip# helper delete -d 37
37: Ok
13:ASN-9000:ip# helper -d
Default UDP helper ports:
42        53        67        68        69
98        137       138
```

5.8.2.5 Clearing Statistics

To clear the IP Helper statistics, issue the **`stats clear helper`** command. Following is an example of the use of this command.

```
12:ASN-9000:ip# stats clear helper
IP helper table stats are cleared.
```

5.8.3 .Setting the Time-To-Live Parameter

The time-to-live (TTL) parameter specifies how long a packet is allowed to remain in the net before it is dropped. Packets that cannot find or are blocked from their destination nodes are dropped when the TTL expires. To change the default TTL, issue the following command:

`ipdefaultttl|ittl set <`*`value`*`>`

<value> Specifies the new TTL time in hops. Specify a number between 1 and 255. The default is 64 hops.

To display the TTL value for outgoing IP packets, issue the following command:

`ipdefaultttl|ittl [show]`

The example below shows how the time-to-live parameter is set and how to display the new setting:

```
46:ASN-9000:ip# ittl set 100
Default TTL is now 100
47:bASN-9000:ip# ittl
Default TTL is 100
```

5.8.4 Enabling and Disabling ICMP Redirect Messages

Use the **`send-icmp-redirect`** command to enable or disable sending ICMP redirect messages by the ASN-9000. In networks that use multiple routers, ICMP redirects messages from routers of alternative routes to segments connected to the routers. Normally, this feature helps optimize routing throughput by ensuring that routers are informed of the most efficient paths to the segments on the network.

The ASN-9000 works well when it receives ICMP redirect messages; however, some other switches do not work well in environments in which these messages are used. If the network contains switches that do not work well when they receive ICMP redirect messages, sending of these messages can be disabled on the ASN-9000.

`send-icmp-redirect|sir enable|disable`

enable|disable Specifies whether ICMP redirect messages are to be enabled or disabled. The default is **`enl`** (enabled).

The example below enables sending of ICMP redirect messages:

```
48:ASN-9000:ip# sir enable
Send ICMP Redirects now enabled.
```

5.8.5 Enabling or Disabling Source-Route Filtering

Use the **`fwd-pkts-with-srcrt-options`** command to disable the source-route feature and strengthen the “firewall” protecting the network from outside users.

IP packets that contain the loose-source-route or the strict-source-route option are forwarded by default. The source-route options are intended to help forwarding of IP packets. When a packet containing a source-route option is forwarded, the packet can appear to receiving devices as though it originated from the device that forwarded it. As a result, these devices are more likely to accept the forwarded packets, rather than filter them.

Disabling the source-route feature prevents outside users from using and exploiting the source-route contained in packets to gain access to the network. The syntax for this command is:

```
fwd-pkts-with-srcrt-option|fps enable|disable
```

enable|disable Specifies whether source-route filtering is to be enabled or disabled. The default is **`enl`** (enabled).

The command below enables source-route filtering:

```
49:ASN-9000:ip# fps enable
Forward Packets with SrcRt Option now enabled
```

NOTE For additional information about IP filtering, see the *ASN*-9000 *Filters Reference Manual.*

5.8.6 Enabling or Disabling Network-Broadcast Forwarding

By default, the ASN-9000 forwards broadcast packets onto subnets attached to the ASN-9000. A network broadcast packet is a packet containing either all zeros or all ones in the host portion of the address. For example: 1.120.255.255, 192.9.200.0, and 10.255.255.255 all are network broadcast packets. The way the software handles broadcast packets differs depending upon how they are received and the destination address specified in the packets.

The ASN-9000 can be forced to forward or drop IP network-broadcast packets sent to subnetted interfaces by enabling or disabling bridge-net-broadcast and route-net-broadcast:

- The bridge-net-broadcast state affects network-broadcast packets received in Ethernet-broadcast packets. If bridge-net-broadcast is enabled, these packets are forwarded. If bridge-net-broadcast is disabled, these packets are dropped.

- The route-net-broadcast state affects network-broadcast packets received in Ethernet-unicast packets. If route-net-bcast is enabled, these packets are forwarded. If route-net-bcast is disabled, these packets are dropped.

The bridge-net-bcast and route-net-bcast states are completely independent of each other. One or both can be enabled or disabled, depending upon the level of broadcast traffic allowed for subnetted interfaces.

IP network-broadcast and IP subnet-broadcast packets can be encapsulated in one of the following types of packets:

- Ethernet-broadcast packets. These packets contain (encapsulate) IP subnet-broadcast packets or IP network-broadcast packets. Ethernet-broadcast packets contain the Ethernet broadcast address (ff-ff-ff-ff-ff-ff) in the destination address field and are received by the ASN-9000 switch from a directly-attached node.
- Ethernet-unicast packets. These packets contain the ASN-9000 switch's Ethernet address in the destination field. Like Ethernet-broadcast packets, Ethernet-unicast packets can contain (encapsulate) IP subnet-broadcast packets or IP network-broadcast packets. However, unlike Ethernet-broadcast packets, Ethernet-unicast packets are received by the ASN-9000 switch from another router.

Forwarding of the following types of IP network-broadcasts can be selectively enabled or disabled:

- IP network-broadcasts sent from a node directly attached to the switch and addressed to a subnetted interface configured on the switch. If bridge-net-bcast is enabled, the packets are bridged to all segments belonging to all subnets in the destination network. If bridge-net-bcast is disabled, the packets are dropped.
- IP network-broadcasts sent from another router and addressed to a subnetted interface configured on the switch. If route-net-bcast is enabled, the packets are routed to all segments belonging to all subnets in the destination network. If route-net-bcast is disabled, the packets are dropped.

Neither the bridge-net-bcast state nor the route-net-bcast state has any effect on IP subnet-broadcast packets or broadcast packets sent to interfaces that are not subnetted if the following conditions are true:

- The interface is subnetted and the received packet is a subnet-broadcast, the packet is unconditionally bridged to all the segments belonging to the same subnet.
- The interface is not subnetted, and the received packet is a net-broadcast packet, the packet is unconditionally forwarded (routed) to all segments in the network.

5.8.6.1 Enabling/Disabling Bridging of Net Broadcasts

To prevent forwarding of IP network-broadcast packets from directly-attached nodes to subnetted interfaces on the ASN-9000, issue the following command:

```
bridge-net-broadcast|bnb disable
```

After this command is issued, network-broadcast packets encapsulated in Ethernet-broadcast packets are still received internally, if applicable, but dropped without being forwarded to the destination subnets. Network-broadcast packets received in Ethernet-unicast packets are not affected.

To re-enable the software to forward network-broadcast packets received in Ethernet-broadcast packets and addressed to subnetted interfaces, issue the following command:

```
bridge-net-broadcast|bnb enable
```

5.8.6.2 Enabling/Disabling Routing of Net Broadcasts

To prevent the forwarding of IP network-broadcast packets from other routers to subnetted interfaces attached to the ASN-9000, issue the following command:

```
route-net-broadcast|rnb disable
```

After this command is issued, network-broadcast packets encapsulated in Ethernet-unicast packets are still received internally, if applicable, but dropped without being forwarded to the destination subnets. Network-broadcast packets received in Ethernet-broadcast packets are not affected.

To re-enable the software to forward network-broadcast packets received in Ethernet-unicast packets and addressed to subnetted interfaces, issue the following command:

```
route-net-broadcast|rnb enable
```

5.8.7 Enabling/Disabling Proxy ARP

The ASN-9000 supports proxy ARP (RFC 1027), a well-defined mechanism in the TCP/IP protocol suite. Using proxy ARP, a router can respond to an ARP request with its own Ethernet address if it knows a route (or default route) to the destination network or subnet on which the requested address resides.

Without proxy ARP, the requesting host needs to have knowledge of its own network, as well as the destination network and the subnet mask, so that it can ARP the destination directly if it is on the same net or ARP the ASN-9000 (or other gateway) if the destination is on a different net.

To use the proxy-arp command to enable or disable the proxy ARP feature for all segments or a specific list of segments, issue the following command:

```
proxy-arp penable|pdisable [<seglist>]
```

<seglist> Specifies the segments on which to enable or disable the feature. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

penable|pdisable Specifies whether to enable or disable the feature. The default is disable.

The following example illustrates the **proxy-arp enable** command for segment 1.23.

```
8:ASN-9000:ip# proxy-arp penable 1.23
Segment 1.23: enabled
```

If a <*seglist*> or **penable|pdisable** is not specified, the current status (enabled or disabled) of the proxy ARP feature is shown.

5.8.7.1 Displaying the Proxy ARP Table

To display the Proxy ARP table, issue the following command:

```
proxy-arp [show] [<seglist>]
```

The following command will display the status for a specific segment:

```
9:ASN-9000:ip# proxy-arp 1.23
IP proxy-ARP status:
Segment 1.23: enabled
```

The **proxy-arp** command will display the status of all segments:

```
71:ASN-9000:ip# proxy-arp show
Segment 2.1: disabled
Segment 2.2: enabled
Segment 2.3: enabled
Segment 2.4: enabled
Segment 2.5: disabled
Segment 2.6: enabled
Segment 2.7: disabled
```

5.9 Showing and Clearing Statistics

The `ip` subsystem maintains statistics on ARP, ICMP, and general IP packets. These statistics are a superset of the corresponding statistics provided in the SNMP MIB. To use the **`stats [show]`** command to display statistics, issue the following command:

```
stats [show] [-t] [-p] [arp|icmp|ip|helper]
```

-t Specifies to display all statistics collected since the software was rebooted, rather than just the statistics collected since the last time the **`stats clear`** command was issued.

-p show statistics per port since last clear (ip only)

[arp|icmp|ip|helper Specifies the type of packet protocol to display statistics.

The ASN-9000 maintains two copies of each IP statistics counter (and similarly for ICMP and ARP packets):

- Count since last clear.
- Count since last switch reset.

Both counters are updated when the corresponding events occur, but the **`stats clear`** command clears only the count since last clear. To display the count since last reset, use the **`-t`** option with the **`stats`** command. To display the count per port since last clear, use the **`-p`** option. Following are some examples of the information displayed by the **`stats`** command. Notice that the first line in each example informs that statistics since the last statistics clear are being displayed, rather than total statistics accumulated since the last reboot.

```
IP statistics: count since last stats clear
Number of Cache Flushes: 1
```

As shown in this example, the IP statistics are organized according to incoming packets and outgoing packets. In addition to totals for packets received, sent, and forwarded, the **`stats ip`** display lists statistics for many of the types of IP routing errors that can occur in a network.

In the following example, ARP statistics are displayed.

```
74:ASN-9000:ip# stats arp
ARP statistics: count since last stats clear
              ARP Packet Statistics:
              Requests received:      38
              Replies received:       25
              Invalid opcodes received:0
              Requests sent:          226
              Replies sent:           36 (0 proxies)
```

Here is an example of the ICMP statistics displayed by the **stats** command.

```
7:bASN-9000:ip# stats icmp
ICMP statistics: Count Since last stats clear
Rcv:                    Echo request:  20
Rcv:                        Messages:  20
Snd:                        Echo rpl:  19
Snd:                    Dest unreach:  58
Snd:            Router Advertisement:  168
Snd:                     TTL expired:  5115
Snd:                        Messages:  5360
Snd:                          Errors:  8
```

5.9.1 Clearing Statistics

The **stats clear** command is used to clear statistics. The syntax for this command is:

stats clear [arp|icmp|ip|helper|all]

[arp|icmp|ip|helper] Specifies the type of packet protocol for which to clear statistics.

all Clears statistics for all.

5.10 Showing or Clearing the IP Route Cache

The IP routing software maintains a route cache containing translation information for the destination hosts. This information is frequently updated based upon incoming packets on each segment. The route cache can be used to determine which hosts are most frequently used.

5.10.1 Displaying the Route Cache

To use the **`cache show`** command to display the route cache, issue the following command:

`cache show [`*<seglist>***`]`**

[<seglist>] Specifies the segments for which to display the route cache. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

5.10.2 Flushing the Route Cache

The route cache can be cleared using the **`cache clear`** command. The **`cache clear`** command removes all entries from the route cache for some or all segments.

After the cache is flushed, new entries are added using the cache’s usual most-recently-used algorithm. If a subsequent **`cache show`** command is issued, fresh entries are displayed.

5.11 Configuring IP Multicast

This chapter describes the IP Multicast commands and how to use them to define IP interfaces on an ASN-9000 as end stations for IP Multicasting. Unlike IP broadcasting, which sends packets to all destinations, or IP unicasting, which sends packets to a single destination, IP Multicasting addresses and delivers packets to a specific subset of destinations.

Using the `ip/mcast` subsystem, IP Multicasting can be set up and used with video conferencing and other multicast applications. The **`ip/mcast`** commands are used to do the following:

- Show the IP Multicast configuration
- Add, show, and delete IP Multicast interfaces
- Add and delete IP Multicast tunnels
- Enable IP Multicast routing
- Add more memory to the IP Multicast route table
- Show and clear the IP Multicast route table
- Show and clear the IP Multicast route cache
- Show and clear IP Multicast statistics
- Enable multicast-aware bridging (for systems that perform IP Multicast routing on VLANs)

5.11.1 Accessing the IP Multicast Subsystem

To access the `ip/mcast` subsystem, issue the following command at the runtime command prompt: **`ip/mcast`**

```
cache                                      neighbors
config                                     pruning
getmem                                     route|rt
interface|it                               stats
ipm                                        transmit
multicast-groups|mg                        tunnel
 multicast-aware-bridging|mab
```

5.11.1.1 Allocating Memory

Before using the `ip/mcast` subsystem, memory must be allocated by issuing the **`getmem`** command, as shown in the following example:

```
1:ASN-9000:ip/mcast# getmem
Memory allocated for IP Multicast.
2:ASN-9000:ip/mcast#
```

If memory has been allocated for IP Multicast at the time the configuration is saved with the **`savecfg`** command, the corresponding **`getmem`** command is placed in the configuration file ahead of other IP Multicast configuration commands. Thus, it is only necessary to type the **`getmem`** command when first configuring the ASN-9000 for IP Multicast routing.

NOTE

FORE Systems recommends that memory for the `ip/mcast` subsystem be allocated immediately after booting to ensure that the memory requested is available. For more information, refer to the *ForeRunner ASN-9000 Installation and Maintenance Manual.*

Memory cannot be de-allocated. To free allocated memory, make sure the configuration file does not contain a **getmem** command, then reboot the software.

5.11.2 Configuring and Displaying IP Multicast Interfaces

A physical interface allows two directly connected ASN-9000s (acting as local routers) to communicate with each other. To define a physical interface, use the **`interface add`** command. The syntax for the command is:

```
it|interface add <ipaddr> [met[ric]<metric>] [thresh[old]<thresh>]
```

<ipaddr> IP address on the local switch, written in dotted-decimal notation. The address must be present in the IP interface table.

[met[ric]<metric> Specifies any additional cost (measured in hops to the destination) of using the interface. The cost range is from 1 through 31. The default is **1**.

[thresh[old]<thresh>] Specifies the minimum time-to-live (TTL) value that an IP Multicast packet must have before it is forwarded over this interface.

This parameter can restrict the types of IP Multicast traffic that go out on a network. The default is **1**.

Following is an example of the use of the **interface add** command:

```
32:ASN-9000:ip/mcast# interface add 192.10.30.33
Okay
33:ASN-9000:ip/mcast#
```

5.11.2.1 Displaying the Interface Table

The **interface [show]** command can be used to display a list of configured virtual interfaces. The display includes both physical interfaces and tunnels. The syntax for this command is:

it|interface [show] [*<disprestrictors>*]

[<disprestrictors>] [[seg[ment[s]]]=]<seglist> Specifies segments for which to display the IP addresses. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

a [ddr[ess]]=<ipaddr> Specifies the IP address.

Following is an example of the interface table displayed by the **interface [show]** command.

```
33:ASN-9000:ip/mcast# it
IP Multicast Routing: virtual interface table:
LocalAddress   RemoteAddressType    SrcRt   Metrc   Thrsh   State   Segments
-------------  -----------------    ----     -----  -----   -----   --------
147.128.70.30  -------------Phy     -----   1       1       Up      2.4,1.8
147.128.128.   -------------Phy     -----   1       1       Up      2.3
147.128.33.5   130.1.5.1    Tunl    No      1       6       Up      2.4,1.8
147.128.30.30  -------------Phy     -----   1       1       Up      1.5
147.128.33.5   192.9.200.21 Tunl    Yes     1       6       Up      1.6
147.128.33.5   ------------- Phy    -----   1       1       Up      1.2
```

This display contains information about four physical interfaces and two tunnels. The tunnel to destination 130.1.5.1 is an encapsulation tunnel. The tunnel to destination 192.9.200.21 is a source-route tunnel.

Local Address and Remote Address Identifies the two ends of a tunnel. The local address corresponds to the configured address for a physical interface.

Type Identifies whether the virtual interface is either a tunnel or a physical interface.

SrcRt Identifies the type of tunnel. Yes in this column indicates that the tunnel is a source-route tunnel. No in this column indicates that the tunnel is an encapsulation tunnel.

Metric Lists the cost (in hops) of the interface.

Thrsh Lists the threshold value for the interface.

State Indicates the state of the interface. Up indicates the interface is active. Down indicates the interface is inactive. The interface is DOWN when a segment from the bridging subsystem is disabled, or if disabled by the automatic segment-state detection mechanism. See your *ForeRunnerASN-9000 Installation and Maintenance Manual* for further information on automatic segment-state detection.

Ports Lists the segments to which the listed virtual interface is assigned.

5.11.2.2 Deleting a Physical Interface

The **`interface del`** command is used to delete a physical interface.

NOTE When a physical interface is deleted, corresponding tunnels are not deleted. To delete a tunnel, use the **`tunnel del`** command.

The syntax for the **`interface delete`** command is:

```
it|interface delete <ipaddr>|all
```

<ipaddr>|all Specifies the IP address of the physical interface to be deleted.

```
35:ASN-9000:ip/mcast# interface del 192.10.30.33
```

If **`all`** is specified, all physical interfaces (excluding the tunnels) are deleted.

5.11.2.3 Enabling Pruning

To enable or disable pruning in the IP Multicast subsystem, issue the following command:

```
pruning enable|disable
```

Following are the results of this command:

```
311:ASN-9000:ip/mcast# pruning enable
IP Multicast pruning enabled.
312:ASN-9000:ip/mcast#
```

5.11.3 Showing the IP Multicast Configuration

The current IP Multicast configuration can be displayed by issuing the **`config show`** command. The syntax for this command is as follows:

```
config [show]
```

Following is an example of the information shown by the **`config show`** command:

```
44:ASN-9000:ip/mcast# show config
   IP Multicast Forwarding: disabled
   Multicast Aware Bridging in a VLAN: disabled
   IPM Pruning: enabled
   Max Routing Entries allocated: 4k

Port State for Multicast Traffic:
Segment  2.1:  Disabled ***
Segment  2.2:  Enabled
Segment  2.3:  Enabled
Segment  2.4:  Enabled
```

In this example, the display produced by the **`show config`** command shows the following information:

- IP Multicast forwarding is enabled.
- Multicast Aware Bridging in a VLAN is disabled.
- IPM pruning is enabled.
- Maximum routing entries allocated is 4k.
- IP Multicast traffic is enabled on all segments except 2.1.

5.11.3.1 IP Considerations

IP Multicast routing works whether IP forwarding is enabled or disabled. In this respect, the ASN-9000 implementation is similar to mrouted, which allows multicast routing on a UNIX workstation even if it is not routing regular IP traffic.

NOTE

IP Multicast routing must be enabled even if the ASN-9000 is configured to have the same subnet on all the segments. The IP Multicast routing code bridges packets intelligently based on reception of membership reports. IP Multicast traffic is restricted to those networks that have listening hosts.

The virtual interface table used for IP Multicast routing is associated closely with the IP interface table. When a virtual interface is added, appropriate information is automatically copied from the IP interface table.

The ASN-9000 updates the segment list in the virtual interface table whenever adding or deleting a segment in an IP interface entry. When an IP interface entry is deleted, all the IP Multicast virtual interfaces that match the deleted entry's address are deleted.

5.11.3.2 Displaying IP Multicast Groups

The **`multicast-groups|mg[show]`** command is used to list the IP Multicast group addresses currently known to the ASN-9000 (acting as the local router). Following is an example of the display produced by this command.

```
35:ASN-9000:ip/mcast# mg
Virtual I\F- : Locaddr: 147.128.70.30  RmtAddr :----, type: Physical
Groups: 224.2.138.32   Segs: 2.1

Virtual  I/F-  Locaddr: ---  RmtAddr:147.128.90.33, type: Tunnel
```

This table contains the list of IP Multicast groups for each virtual interface and contains the following information:

Locaddr Displays additional statistics, including the number of packets and octets transmitted to and received from the net by each interface.

RmtAddr Lists the IP address of a remotely attached IP Multicast neighbor. This applies only to tunnels, in which the ASN-9000 and the other end of the virtual interface are separated by gateways.

type Lists the type of IP Multicast interface. Valid types are `Physical` and `Tunnel`.

Groups Lists the IP Multicast groups. The group IP address and the ASN-9000 segment(s) on which membership reports for that group were received are listed for each group.

5.11.3.3 Displaying IP Multicast Neighbors

The **`neighbors`** [**`show]`** command is used to list all the neighboring routers currently known. Following is an example of the display produced by this command.

```
35:ASN-9000:ip/mcast# neighbors
Virtual I\F- :Locaddr: 147.128.128.99 RmtAddr :----,type:Physical,
Neighbors: 147.128.128.30   (25 sec)   147.128.100.2  (40 sec)
Virtual  I/F-  Locaddr:  147.128.128.99 Rmtaddr---,type:Tunnel,
Neighbors: 130.1.5.1   (35 sec)
```

This display contains a list of neighboring routers for each virtual interface and contains the following information:

Locaddr Lists the IP address of a directly-attached IP Multicast neighbor. This applies only to physical interfaces, in which the ASN-9000 and the other end of virtual interface are directly attached.

RmtAddr Lists the IP address of a remotely attached IP Multicast neighbor. This applies only to tunnels, in which the ASN-9000 and the other end of the virtual interface are separated by gateways.

type Lists the type of IP Multicast interface. Valid types are `Physical` and `Tunnel`.

Neighbors Lists the IP Multicast neighbors. The router's IP address and the number of seconds elapsed since the last routing update was received from this neighbor is listed for each neighbor.

5.11.4 Configuring and Displaying Tunnels

A tunnel is a type of virtual interface that allows theASN-9000 (acting as the local router) to communicate with a remotely attached router.

5.11.4.1 Adding a Tunnel

The **`tunnel add`** command is used to define a tunnel. The syntax for this command is:

```
tunnel add [-s] loc[al]<local-addr> rem[ote]<remote-addr>
            [met[ric]<mv>] [thresh[old]<tv>]
```

[-s] Specifies that the tunnel is a source-route tunnel, rather than an encapsulation tunnel.

If **`-s`** is not specified, this command automatically configures the tunnel as an encapsulation tunnel.

loc[al]<localaddr> Specifies the IP address of the local ASN-9000. The address must be one of the configured IP addresses listed in the IP interface table.

rem[ote]<remoteaddr> Specifies the IP address of the router at the other end of the tunnel.

[met[ric]<mv> Specifies an additional cost (extra hops to the destination) of using the virtual interface with which this tunnel is associated. Specify a number in the range 1 through 31. The default is **`1`**.

[thresh[old]<tv> Specifies the minimum time-to-live (TTL) value that an IP Multicast packet must have before it can be forwarded through the tunnel. This parameter restricts IP Multicast datagrams from going out on a network. The default is **`1`**.

Following is an example of how to add a tunnel. In this example, the **`-s`** argument is not used, so the software creates an encapsulation tunnel. The default values are accepted for the metric and threshold.

```
34:ASN-9000:ip/mcast# tunnel add loc 192.10.30.33 rem 155.10.23.222 met 3 thresh 4
Okay
```

5.11.4.2 Deleting a Tunnel

The **`tunnel del`** command is used to delete a virtual interface that maps to a tunnel. The syntax for this command is:

```
tunnel del (loc[al]<local-addr> rem[ote]<remote-addr>)|all
```

loc[al]<localaddr> Specifies the IP address of the ASN-9000 (the local end of the tunnel).

rem[ote]<remoteaddr> Specifies the IP address of the router at the remote end of the tunnel.

The following command deletes a tunnel:

```
35:ASN-9000:ip/mcast# tunnel add loc 192.10.30.33 rem 155.10.23.222 met 3 thresh 4
```

To delete all IP Multicast tunnels, issue the following command:

```
tunnel del all
```

5.11.5 Enabling IP Multicast Routing

To enable IP multi-cast routing, use the **`enable ipm`** command:

```
enable|disable ipm
```

enable|disable Specifies whether you are enabling or disabling IP Multicast forwarding. The default is **`disable`**.

NOTE If IP Multicast forwarding is enabled immediately after enabling RIP listening, the multicast route updates are not accepted over a tunnel until the IP routing table learns either an entry to the remote end of the tunnel or a default route.

5.11.5.1 Enabling Multicast Traffic on a Segment

IP Multicast forwarding can be restricted on a segment-by-segment basis. The syntax for the command used to enable or disable IP Multicast traffic on a set of segments is shown below.

transmit penable|pdisable *<segment-list>*

penable|pdisable Specifies whether IP Multicast forwarding is to be enabled or disabled. The default is **penable**.

<segment-list> Specifies the list of segments on which IP Multicast forwarding is being enabled or disabled. If **all** is specified, IP Multicast forwarding is enabled or disabled on all segments.

The first command in the following example uses the **transmit** command to enable IP Multicast traffic on segments 2.4. The second command uses the **set** command to disable IP Multicast traffic on segment 2.2.

```
46:ASN-9000:ip/mcast# transmit penable 2.4
Ok
47:ASN-9000:ip/mcast# transmit pdisable 2.4
Ok
```

5.11.6 Configuring and Displaying IP Multicast Routes

The **route show** command is used to display a list of IP addresses originating IP Multicast traffic, currently known by the IP Multicast routing software. The syntax for this command is:

route|rt [show] [-c|-r] [-d] [-t] [-s] [*<disprestrictors>*]

[-c|-r] **-c** displays directly connected routes only.
-r displays Distance Vector Multicast Routing Protocol (DVMRP) routes only.

[-d] Displays the routing table in detail.

[-t] Displays the total number of routes only.

[-s] Displays the output in sorted order.

[<disprestrictorst>] [[seg[ment[s]]]=]<seglist> Specifies the ASN-9000 segments for which to display route information.

a[ddr[ess]]=<ipaddr> Specifies the IP address (origin) of the route entries to be displayed.

Following is an example of the display produced by the command.

```
52:ASN-9000:ip/mcast# route show
IP Multicast Routing table:
Origin         Origin Mask    Gateway          Met  Age   Parent. Segs/Children
-----------    -------------- -------------    ---  ---   ----   -------
147.128.70.0   255.255.255.0  --------------   1    ---   ----   4.1
147.128.128.0  255.255.255.0  --------------   1    ---   ----   2.2
147.128.90.0   255.255.255.0  --------------   1    ---   ---    5.1,6.1
129.155.80.0   255.255.240.0  147.128.70.2     3    20    4      2.5,6.2
150.233.0.0    255.255.0.0    47.128.128.111   5    35    2      4.5,2.6
```

The route table contains the following information:

Origin Lists the IP address of the origin network. An origin is a network that is capable of originating IP Multicast traffic.

Origin Mask Lists the origin mask used on the origin network. An origin mask is the subnet mask of an origin network.

Gateway Lists the IP address of the next-hop router to the origin. This column is not applicable to directly connected entries.

Met Displays the total cost (or metric) of reaching the origin. This metric is the sum of the cost of the next-hop virtual interface and the number of hops or intervening routers (if applicable) used to reach the origin.

Age Shows the time elapsed, in seconds, since a DVMRP route report was last received for this origin. This column is not applicable to directly connected routes.

Parent Shows the segment on which the next-hop router is located for a dynamic route. This column is not applicable to directly connected routes.

Segs/Children For directly-connected routes, the Seg/Children column shows the segments on which the corresponding virtual interface is configured. For a dynamic entry, this column lists the segments on which the IP Multicast packets from this origin are forwarded.

5.11.6.1 Clearing the Route Table

The **`route clear`** command is used to flush all dynamically learned entries from the route table. The example below shows this command:

```
66:ASN-9000:ip/mcast# route clear
Okay
```

5.11.6.2 Adding Memory

Additional memory can be allocated to the IP multicast routing table by using the **`addmem`** command. The parameter allows specification of 2k, 4k, and 6k routes. Enter the command as shown below:

`addmem <2k|4k|6k>`

```
66:ASN-9000:ip/mcast# addmem 2k
```

5.11.7 Using the IP Route Cache

The IP Multicasting software maintains a route cache containing translation information for the destination hosts. This information is frequently updated based upon incoming packets on each segment. The route cache can be used to determine which hosts are most frequently used. Because the contents of the route cache can change rapidly, successive **`cache show`** commands can give different results.

5.11.7.1 Displaying and Clearing the Route Cache

The **`cache show`** command is used to display the route cache. The syntax for this command is:

`cache [show]`

The route cache can be flushed (cleared) using the **`cache clear`** command. This command removes all entries from the route cache for some or all segments. After the cache is flushed, new entries are added using the cache's usual most-recently-used algorithm. If a subsequent **`cache show`** command is issued, fresh entries are displayed.

5.11.8 Displaying Statistics

The `ip/mcast` subsystem maintains statistics on DVMRP, Internet Group Management Protocol (IGMP) and routed packets. To display statistics, issue the following command:

```
stats [show] [-t] [dvm|igmp|rt|all]
```

[-t] Displays statistics collected subsequent to the last system reset, rather than merely the last time statistics were cleared.

[dvm|igmp|rt|all] Displays the type of packet for which statistics are desired:

dvmDisplays DVMRP packet statistics.

igmpDisplays IGMP packet statistics.

rtDisplays routing packet statistics.

Following is an example of the display produced by the **`stats show dvm`** command, used to display DVMRP statistics.

```
5:ASN-9000:ip/mcast# stats show dvm
DVMRP Statistics (count since last stats clear):
Route reports sent:                     32
Neighbor probes sent:                   0
Neighbor prunes sent:                   0
Neighbor grafts sent:                   0
Neighbor graft_acks sent:               0
Neighbor responses sent:                12
Neighbor2 responses sent:               0

Route reports received:                 211
Neighbor probes received:               1
Neighbor prunes received:               0
Neighbor grafts received:               0
Neighbor graft_acks received:           0
Neighbor requests received:             0
Neighbor2 requests received:            33

Rcvd pkts with bad metric:              1
Rcvd pkts with bad orig. mask:          0
Rcvd conflicting route reports:         0
Rcvd truncated route reports:           0
Conflicting routes deleted:             0
Rcvd reports from non neighbor:         5
Rcvd probes from non neighbor           5
Rcvd prunes from non neighbor           5
Rcvd grafts from non neighbor           5
Rcvd graft_acks from non neighbor       0
Rcvd invalid neighbor requests:          0
Rcvd invalid neighbor responses:         0
Rcvd invalid Neighbor2 responses:        0
Rcvd message from non neighbor:          0

No mem to receive packet:                2
No memory to send packets:               0
```

Following is an example of the display produced by the **`stats show igmp`** command, used to display IGMP statistics.

```
60:ASN-9000:ip/mcast# stats show igmp
IGMP Statistics (count since last stats clear):
total packets received:                      551
short packets received:                      2
pkts rcvd with checksum error:               0
total membership queries rcvd:               12
invalid membership queries rcvd:             0
total membership reports rcvd:               333
invalid membership reports rcvd:             0
rcvd packets too big:                        0
rcvd unknown DVMRP message:                  0
rcvd unknown IGMP message:                   0
packets looped back:                         9
no buffer for looping back:                  0
no timers for multicast routing:             0
report not sent - no interface:              0
group timer not started - no I/F:            0
rcvd report from non adj. host:              1
total membership queries sent:               9
total packets sent:                          159
total packets not sent:                      0
no memory to process rcvd pkts:              2
Queue blocks accessed:                       2
Queue blocks released:                       2
Free Queue blocks available:                 2048
```

Below is an example of the display produced by **stats show rt** command which displays routing statistics.

```
59:ASN-9000:ip/mcast# stats rt
Multicast routing statistics (count since last clear):
route cache hits:                     661
route cache misses:                   661
route cache flushed:                  0
route lookups:                        661
route lookups misses:                 661
source group pair cache lookups:      11322
source group pair cache misses:       11322
rcvd msg over invalid tunnel:         5
no room for tunnel options:           0
rcvd msg on wrong interface:          17
packets forwarded:                    3213
packets dropped:                      2448
packets received:                     5661
rcvd packet format error:             0
encapsulated packets rcvd:            2112
rcvd port not configured:             0
no route to origin:                   2448
packets bridged:                      1123
packets not bridged:                  0
no memory to process packets:         0
```

5.11.8.1 Clearing Statistics

The **stats clear** command is used to clear statistics for DVMRP, IGMP, or route packets. The syntax for this command is:

```
stats clear dvm|igmp|rt|all
```

5.11.9 Enabling Multicast-Aware Bridging

The ASN-9000 supports VLANs for IP routing. A VLAN is an IP interface configured on multiple segments. When the ASN-9000 receives a packet on an IP Multicast virtual interface that maps to multiple physical segments, it can bridge the packet to other segments and simultaneously route it to other virtual interfaces, transmitting the same copy of the packet on all segments. When this occurs, the time-to-live (TTL) of the bridged packets, as well as the routed packets, is reduced by one. Because this procedure avoids copying the packet again, it results in improved performance. Because most multicast applications use a large TTL value, a one-hop reduction when bridging occurs should not significantly affect performance.

If it is not desired that IP Multicasting make its forwarding decisions when it receives membership reports on a port, Multicast bridging can be disabled by issuing the following command:

```
multicast-aware-bridging|mab disable
```

To re-enable Multicast bridging, use the multicast-aware-bridging enable command:

```
multicast-aware-bridging|mab enable
```

The default is disabled.

5.12 Configuring IP/RIP

In a routed environment, routers communicate with each other to keep track of available routes. The ASN-9000 routing software implements standard Routing Information Protocol (RIP) for exchanging TCP/IP route information with other routers. RIP uses User Datagram Protocol (UDP), an industry-standard connectionless protocol, for sending and receiving packets between the ASN-9000 and other devices. The RIP protocol allows the use of RIP version 1 and version 2 routes on the same subnet. However, since RIP version 1 routers are not capable of address subnetting, having RIP version 1 routers with RIP version 2 routers can lead to network problems. It is strongly recommended that all RIP routers on a subnet run the same RIP version.

This chapter describes how to use `ip/rip` subsystem commands to perform the following tasks:

- Display the RIP configuration
- Configure RIP parameters for IP networks
- Display and clear RIP statistics
- Enable RIP Bridging (used only when IP traffic crosses the ASN-9000 on a VLAN)

5.12.1 Accessing the RIP Subsystem

To access the `ip/rip` subsystem, issue the following command at the runtime command prompt: **`ip/rip`**

```
AccptDefaultRt|ad                     rxtype
auth                                  splitHorizon|sph
authtype                              stats
backup-route|br                        talk|ta
config|conf                            tracesettings|tr
interface|it                           tracelevel|trl
neighbor|nbr                           traceclass|trc
listen|li                              txtype
metric                                 filter
poison|po                              template
rip
```

5.12.2 Displaying the RIP Configuration

The configuration command shows the configuration of the IP RIP subsystem. If ifaddr is specified, information about this interface is shown. With no parameters, information about all RIP interfaces is displayed, global RIP configuration settings, as well as the configuration setting for all RIP interfaces. The syntax for this command is:

`config|conf [show] [<ifaddr>]`

Entering **conf** from the ip/rip subsystem displays the following information:

```
38:ASN-9000:ip/rip# conf

                           -- RIP Configuration --

RIP routing: enabled

Keeping backup routes: disabled

I/F Addr     Tx    Rx    PoisonSplitAcpt Auth  TxtypeRxtypeMetricAdmin Oper
                              Horiz Def
------------------ ---   ----------  ---- ----  ---------------- ----- --
169.144.86.54 yes  yes    no  yes   yes  off    rip2 both  1    up     up
```

where

I/F Addr The IP address of this RIP interface

Tx Displays “yes” if this interface is sending updates. Otherwise “no” is displayed.

Rx Displays “yes” if this interface is receiving updates. Otherwise “no” is displayed.

Poison Displays “yes” if poison reverse processing is enabled on this interface. Otherwise “no” is displayed.

SplitHoriz Displays “yes” if splithorizon processing is enabled on this interface. Otherwise “no” is displayed.

AcptDef Displays “yes” if default route is accepted in an update on this interface. Otherwise “no” is displayed.

Auth Displays “on” if this interface sends out and expects authentication updates. Otherwise “off” is displayed.

Txtype RIP version of updates sent from this interface.

Rxtype RIP version of updates accepted on this interface.

Metric Cost associated with this interface. This cost is added to every incoming route.

Admin The administrative status of this interface.

Oper The operational status of this interface.

5.12.2.1 Interface

The **`interface`** command is used to add, delete, enable, disable, or show RIP interfaces. By default all configured interfaces are displayed when executed without parameters. The display can be restricted to show interface parameters for a specified interface address. The syntax for this command is:

```
it|interface add <ifaddr>
it|interface delete <ifaddr>
it|interface enable <ifaddr>
it|interface disable <ifaddr>
it|interface [show] [<ifaddr>]
```

where

add Adds and enables a RIP interface on a specified interface address (<ifaddr>).

delete Disables and deletes a RIP interface on a specified interface address (<ifaddr>).

enable Administratively enables a RIP interface on a specified interface address (<ifaddr>).

disable Administratively disables a RIP interface on a specified interface address (<ifaddr>).

[show] Displays information about all configured RIP interfaces or about the specified interface (<ifaddr>), if supplied.

Entering **`interface`**, without parameters, displays the following:

```
56:ASN-9000:ip/rip# interface
       -- RIP Configuration --

I/F Addr     Tx   Rx    PoisonSplitAcpt  Auth  TxtypeRxtypeMetricAdmin Oper
                              Horiz Def
------------------ ---   ---------- ---- ---- ----------------- ----- --
169.144.86.54 yes  yes    no  yes   yes  off    rip2 both  1    up     up
```

where

I/F Address The IP address of this RIP interface.

Tx Displays “yes” if this interface is configured to send updates. Otherwise “no” is displayed.

Rx Displays “yes” if this interface is configured to receive updates. Otherwise “no” is displayed.

Poison Displays “yes” if poison reverse processing is enabled on this interface. Otherwise “no” is displayed.

SplitHoriz Displays “yes” if split horizon processing is enabled on this interface. Otherwise “no” is displayed.

AcptDef Displays “yes” if default route accepted in an update is configured on this interface. Otherwise “no” is displayed.

Auth Displays “on” if this is interface is configured to send out and expects authentication updates. Otherwise “off” is displayed.

Txtype RIP version of updates sent from this interface.

Rxtype RIP version of updates accepted on this interface.

Metric Cost associated with this interface. This cost is added to every incoming route.

Admin The administrative status of this interface.

Oper The operational status of this interface.

5.12.2.2 RIP Routing

The **`rip`** command can be used to enable / disable RIP routing on the ASN-9000. Disabling RIP routing results in all routed learned through RIP to be cleared from the routing table. The syntax for this command is:

```
[rip] enable
[rip] disable
```

5.12.2.3 Backup Route

The **`backup-route`** command is used to enable or disable backup routes. The syntax for this command is:

```
backup-route|br enable
backup-route|br disable
```

where

enable Enables holding of RIP backup routes in the routing table. When enabled, up to two best routes per destination are kept in the routing table. The second route appears as a “backup route” in the output of the `ip` **`route show`** command.

disable Disables holding of RIP backup routes in the routing table.

NOTE

For networks with loops, backup routes decrease the convergence time in case of a change in network topology. In these cases, it is strongly recommended that backup-routes be disabled.

5.12.2.4 Neighbor

The **`neighbor`** command is used to add or delete trusted neighbors to a specified interface address or to show both configured and discovered neighbors on a RIP interface or all configured RIP interfaces. The display can be restricted to show either neighbors on a specific interface or by the neighbor type, configured or discovered. If trusted neighbors are added to a RIP interface, only updates from these neighbors are processed. The syntax for this command is:

```
neighbor|nbr [n]add <ifaddr> <nbraddr>
neighbor|nbr [n]delete <ifaddr> <nbraddr>
neighbor|nbr [show] [-c | -d] [<ifaddr>]
```

where

[n]add Add (trusted) RIP neighbor <nbraddr> to the specified interface <ifaddr>.

[n]delete Delete (trusted) RIP neighbor <nbraddr> from the specified interface <ifaddr>.

[show] Display all neighbors, or those neighbors on specified interface address (<ifaddr>). The display can be filtered to display:
-c = Show only configured neighbors.
-d = Show only discovered neighbors.

Entering **`nbr`** from the `ip/rip` subsystem displays the following:

```
106:ASN-9000:ip/rip# nbr

               --- Trusted Neighbors ---

I/F Addr       NbrAddress     Type           Last heard(sec)
----------------------------------------     -----------------
169.144.86.54  169.144.86.49  discovered     5

107:ASN-9000:ip/rip#
```

where

I/F Addr Interface address associated with this neighbor.

Nbr Address Trusted neighbor address associated with this interface.

Type Displays whether this particular neighbor was discovered or configured.

Last Heard (sec) Displays, in seconds, when this particular trusted neighbor was last heard from.

5.12.2.5 Metric

The **metric** command sets the cost associated with an interface. This value is added to every route learned through this interface. The syntax for this command is:

```
metric [n]set value(1-15) <ifaddr>
```

```
43:ASN-9000:ip/rip# metric set 5 169.144.86.54
metric set to 5 on interface 169.144.86.54
```

5.12.2.6 Split Horizon

The **splitHorizon|sph** command is used to avoid problems caused by including routes in updates sent to a gateway from which they were learned. Split horizon omits routes learned from one neighbor in updates sent to that neighbor. Split horizon with poisoned reverse includes such routes in updates, but sets the metrics to infinity. The syntax of this command is:

```
splitHorizon|sph [n]enable <ifaddr>
splitHorizon|sph [n]disable <ifaddr>
```

5.12.3 Trace Settings, Trace Level, and Trace Class Commands

The trace settings, trace level, and trace class commands are used for debugging purposes only.

5.12.3.1 Trace Settings

The **tracesettings|tr** command displays the current trace settings. Trace settings include trace levels and enabled trace classes. By default non-interface related and trace settings for all interfaces are shown. The display can be restricted to a specific interface address (<ifaddr>). The syntax for this command is:

```
tracesettings|tr [show] [<ifaddr>]
```

Issuing **tracesettings** displays:

```
4:ASN-9000:ip/rip# tracesettings
                       -- Trace settings --
Entity             Action     Level      Enabled classes
---------------    -------    ---------- --------------------------
RIP:
                   Print      (warning)  gen timer route txpkt rxpkt
Intf: 169.144.86.54
                   Print      (warning)  gen timer route txpkt rxpkt
```

5.12.3.2 Trace Level

The **tracelevel|trl** command sets trace levels of info, notice or warning on a specified RIP interface address (<ifaddr>) or all RIP interfaces if no interface address is specified. Entering no parameters displays the current trace level settings of all configured RIP interfaces. The syntax for this command is:

```
tracelevel|trl [n]set level-name [<ifaddr>]
```

The example below sets the trace level to warning for all configured RIP interfaces:

```
21:ASN-9000:ip/rip# trl set warning
```

Entering **tracelevel** from the `ip/rip` subsystem displays the following information:

```
20:ASN-9000:ip/rip# trl
               -- Trace settings --

Entity         Action Level   Enabled classes
---------------------- ------------------------------------
RIP:
               Print  (warning)gen timer route txpkt rxpkt
Intf: 169.144.86.54
               Print  (warning)gen timer route txpkt rxpkt

21:ASN-9000:ip/rip#
```

5.12.3.3 Trace Class

The **traceclass|trc** command enables or disables displaying trace information for classes rxpkt, txpkt, route, and gen. Entering **traceclass** with no parameters displays the current trace settings. The syntax for this command is:

```
traceclass|trc [n]enable class-name [<ifaddr>]
traceclass|trc [n]disable class-name [<ifaddr>]
```

The example below disables the route class on all configured interfaces:

```
24:ASN-9000:ip/rip# trc disable route
```

Entering **traceclass** from the `ip/rip` subsystem displays the current settings:

```
25:ASN-9000:ip/rip# trc
           -- Trace settings --
Entity         Action Level      Enabled classes
---------------------- ---------- --------------------------
RIP:
               Print   (notice)    gen timer txpkt rxpkt
Intf: 169.144.86.54
               Print    (notice)    timer route txpkt rxpkt
```

5.12.3.4 Authentication

The authentication command shows the current authentication settings on a configured RIP interface. By default, authentication settings on the RIP interfaces are displayed. The display can be restricted to display settings for RIP interfaces on a specific interface address. Additionally, the **auth** command is used to enable, disable, set and unset authentication parameters. The syntax for this command is:

```
auth [n]enable <ifaddr>
auth [n]disable <ifaddr>
auth [n]set [-k <keyid>|<password>] <ifaddr>
auth [show] [ifaddr]
auth [n]unset <ifaddr>
```

where

[n]enable Enables authentication of RIP updates sent from/to the specified interface <ifaddr>.

[n]disable Disables authentication of RIP updates sent from/to the specified interface <ifaddr>.

[n]set Sets authorization string or key identifier on the specified interface <ifaddr>. Maximum length of <password> is 16 characters, <keyid> must be in the range of 0-255.

[n]unset Unsets (clears) the authorization string on the specified interface <ifaddr>.

[show] Displays authentication related information for the specified interface <ifaddr> or all interfaces if none is specified.

Entering **auth** from the ip/rip subsystem displays:

```
70:ASN-9000:ip/rip# auth
              --- Interface authentication ---
  I/F Addr      State  Type  passwd/keyid
--------------- ----- ------ ----------------
169.144.86.54   off     none   ---
71:ASN-9000:ip/rip#
```

where

I/F Addr Interface address.

State Displays whether authorization for this interface is set (on) or unset (off).

Type Displays the type of authorization, either simple or md5 related.

password/keyid Password or keyid associated with this interface address authentication.

5.12.3.5 Setting, Enabling, and Disabling Authentication of RIP Updates

1. To enable authentication of RIP Version 2 updates sent to the network, use the auth enable command.

   a. Before enabling authentication, set the password and key id. To set the keyid, go the nvram subsystem. The example below shows how to set the keyid. For more information, refer to chapter 8 in the *ForeRunner ASN-9000 Software Reference Manual*.

```
10:ASN-9000::nvram# md5key[45] set 23456
```

   b. Then set the authentication in the ip/rip subsystem, using the auth set command:

```
auth [n]set [-k <keyid>|<password>] <ifaddr>
```

```
14:ASN-9000:ip/rip# auth set -k 45 147.128.9.7
```

2. After the keyid and the password have been set, enable authentication using the **auth enable** command:

```
auth [n]enable <ifaddr>
```

-k<keyid> Specifies the value to be used as the Authentication Key that has a simple password value. If a string shorter than 16 octets is supplied, the string is left-justified and padded to 16 octets, on the right, with nulls (0x00).

<password> Specifies simple password. For a simple password, specify any combination of up to eight numbers, letters, and special characters.

none|nSpecifies that the OSPF area being added does not use authentication.

simple-password|spSpecifies that a password is required for OSPF packets sent within this area.

md5|mSpecifies that MD5 authentication is required for OSPF packets sent within this area. See RFC 1321 for information about MD5 authentication.

<ifaddr> Specifies the IP interface address for which authentication of RIP Version 2 updates is to be enabled. Specify a specific IP interface address or a comma-separated list of addresses.

```
25:ASN-9000:ip/rip# auth nenable 169.144.86.54
```

3. To disable authentication of RIP Version 2 updates sent to the network, issue the following command:

```
auth [n]disable <ifaddr>
```

5.12.3.6 Setting the Authentication String on an Interface

To set an authorization string or a key identifier on a specified VLAN, issue the following command:

```
auth [n]set [-k <keyid>|<password>] <ifaddr>
```

<ifaddr> Specifies the IP interface address for which to set an authorization sting or a key identifier on a specified VLAN. Specify a specific IP interface address or a comma-separated list of addresses.

To unset an authorization string or a key identifier on a specified VLAN, issue the following command:

```
auth nunset [-k <keyid>|<password>] <ifaddr>
```

<ifaddr> Specifies the IP interface address for which to unset an authorization sting or a key identifier on a specified VLAN. Specify a specific IP interface address or a comma-separated list of addresses.

5.12.3.7 Setting the Receive and Transmit Type on a VLAN

To set the receive type on a specified VLAN, issue the following command:

```
rxtype nset rip1|rip2|both <ifaddr>
```

rip1|rip2|both Specifies the receive type of RIP-1 packets, RIP-2 packets, or both RIP-1 packets, RIP-2 packets on a VLAN.

<ifaddr> Specifies the IP interface address for which to set receive type on a specified VLAN. Specify a specific IP interface address or a comma-separated list of addresses.

The example below sets the receive type to RIP-1 packets on the specified interface address:

```
23:ASN-9000:ip/rip# rxtype set rip1 169.144.86.54
rxtype set to rip1 on interface 169.144.86.54
```

To set the transmit type on a specified VLAN, issue the following command:

```
txtype nset rip1|rip1c|rip2 <ifaddr>
```

rip1|rip1c|rip2 Specifies the following type of RIP packets to be transmitted on a VLAN:

rip1Specifies that RIP-1 messages are sent.

rip1cSpecifies that RIP-2 messages are sent broadcast.

rip2Specifies that RIP-2 messages are sent multicast.

<ifaddr> Specifies the IP interface address for which to set transmit type on a specified VLAN. Specify a specific IP interface address or a comma-separated list of addresses.

The example below sets the transmit type to Rip1c, specifying that RIP-2 messages are sent broadcast, on the specified interface address:

```
25:ASN-9000:ip/rip# txtype set rip1c 169.144.86.53
txtype set to rip1c on interface 169.144.86.53
```

5.12.4 Displaying and Clearing RIP Statistics

The `rip` subsystem maintains statistics on RIP packets that it transmits and receives. The **`stats`** command is used to display statistics. Statistics accumulated since the last system reset, or since the most recent statistics clear can be displayed. The syntax for the **`stats show`** command is:

`stats [show] [-t]`

-t Displays statistics accumulated since the last switch reset. If this argument is not used, the statistics accumulated since the last statistics clear are displayed.

Following is an example of the information displayed by this command:

```
26:ASN-9000:ip/rip# stats

RIP Packet Statistics (Total count since last stats clear)
-----------------------------------------------------------------------
                      Pkts Rcv:     12
                     Pkts Sent:     1072
                 Requests Rcvd:     0
                Responses Rcvd:     12
                 Requests Sent:     3
                Responses Sent:     1069
                Route Timeouts:     115
            Bad Size Pkts Rcvd:     0
                 Bad Vers Rcvd:     0
                Bad Zeros Rcvd:     0
              Bad SrcPort Rcvd:     0
                Bad SrcIP Rcvd:     0
                Pkts From Self:     0
```

Use the **`stats clear`** command to clear statistics. As soon as this command is issued, the ASN-9000 clears the counters for statistics collected since the last statistics clear. Statistics accumulated since the last reboot are not cleared.

5.13 Configuring IP/OSPF

This section lists the ASN-9000 requirements for using Open Shortest Path First (OSPF) and describes basic features of OSPF. For complete information about OSPF, refer to RFC 2178. The ASN-9000 implementation of OSPF is based on this RFC.

5.13.1 Accessing the IP/OSPF Subsystem

To access the `ip/ospf` subsystem, issue the following command at the runtime prompt:

```
ip/ospf
```

Listed below are the commands available at this level:

```
4:ASN-9000:ip/ospf# ?
ip ospf subsystem:
area|ar                               net-range|nr
asbd                                  ospf
config|conf                           stats
interface|it                          template
filter                                virtual-link|vlink
getmem                                tracesettings|tr
lsdb                                  tracelevel|trl
external-lsdb|elsdb                   traceclass|trc
neighbor|nbr
```

5.13.2 Configuring an ASN-9000 Switch as an OSPF Router

The ASN-9000 can be configured as the following types of OSPF router:

- Internal
- Backbone
- Area Border
- Autonomous System Border

An OSPF router can function as more than one of the router types listed above. For example, a system that has interfaces attached to the backbone and to other OSPF areas can function both as a Backbone router and as an Area Border router.

Generally, it is not necessary to worry about the differences among these router types. The OSPF software determines how the ASN-9000 is being used based upon the network configuration. Unless OSPF areas are configured using the **`area|ar add`** command, the ASN-9000 assumes that the system is configured as a Backbone router. In addition, the software automatically configures area ID 0.0.0.0 for the backbone.

To configure the ASN-9000 for OSPF routing, perform the following tasks. These tasks apply to all OSPF router types.

- Allocate memory for OSPF.
- Add IP interfaces (if interfaces are not already configured) see section 5.2 of this chapter. Enable IP forwarding (if not already enabled) *see section 5.2*.
- Assign the OSPF router ID.

Depending upon the type of OSPF router to be used, it may be necessary to perform some additional configuration tasks.

- If the ASN-9000 is to be used as an Interior router or an Area Border router, add OSPF areas, then add OSPF interfaces to the areas.
- If the network contains areas that are not connected to the backbone and are not connected to each other, and the Area Border router for one of these areas is not a ASN-9000, it may be necessary to create virtual links.
- If the ASN-9000 is to be used as an Autonomous System Border router, enable the ASN-9000 as this type of router.

Finally, after completing the OSPF configuration steps listed above, enable OSPF routing. The following sections describe how to perform these tasks.

5.13.2.1 Allocating Memory

A portion of main memory must be allocated for the `ospf` subsystem. It cannot be accessed if memory is not allocated. To allocate memory for the `ospf` subsystem, issue the following command:

```
getmem
```

5.13.2.2 Enabling/Disabling OSPF

To enable OSPF, issue the following command:

```
ospf enable
```

To disable OSPF, issue the following command:

```
ospf disable
```

5.13.2.3 Configuration

The OSPF configuration command, **config|conf,** has been modified to remove the Automatic Virtual Link Feature. The Router ID command can now be found in the IP subsystem. The syntax for this command is:

config|conf [show|sh]

Entering **config** from the `ip/ospf` subsystem displays:

```
91:ASN-9000:ip/ospf# config
OSPF Router                                 :memory available
OSPF Routing                                :Disabled
OSPF Router ID                              :169.144.86.54
OSPF Area Border Router                     :No
OSPF Autonomous System Boundary Router:     :Disabled
92:ASN-9000:ip/ospf#
```

5.13.2.4 Assigning the OSPF Router ID

Each OSPF router within the Autonomous System must have a unique OSPF router ID. The OSPF router ID is a 32-bit address in IP format. The software does not assign an address automatically.

Any 32-bit address can be used for the OSPF router ID. However, FORE Systems recommends that one of the IP addresses configured on the ASN-9000 be used. Using one of the IP addresses on the ASN-9000 ensures that OSPF IDs remain unique. If an IP address configured on the switch is chosen, this does not affect IP or OSPF. That is, the software does not establish a special relationship between the IP address chosen and the OSPF software.

By requiring that an IP address configured on the switch be used, the ASN-9000 OSPF software ensures that the OSPF router ID remains unique regardless of changes in the network. To assign the OSPF router ID, issue the following command:

router-id set *<router-id>*

<router-id> Specifies the OSPF router ID in dotted decimal notation (xxx.xxx.xxx.xxx, where each “x” is an integer from 0 through 9).

NOTE

Define the OSPF router ID only when OSPF routing is disabled. To verify that OSPF routing is disabled, issue the **config show** command.

Following is an example of this command:

```
2:ASN-9000:ip/ospf# router-id set 1.1.1.1
```

5.13.2.5 Displaying the Router-ID

To display the router-id table, issue the following command:

```
router-id [show]
```

Following is an example of this command:

```
11:ASN-9000:ip/ospf# router-id

OSPF Router                              : memory available
OSPF Routing                             : Disabled
OSPF Router ID                           : 1.1.1.1
OSPF Version number                      : 2
OSPF Area Border Router                  : No
OSPF Autonomous System Boundary Router  : Enabled
```

5.13.2.6 Adding an OSPF Area

When OSPF routing is enabled, the ASN-9000 automatically creates an OSPF area for the network backbone. The area ID for the backbone is always 0.0.0.0. Depending upon how the network is organized, additional OSPF areas may need to be added. To add an OSPF area to the ASN-9000, issue the following command:

```
area|ar add <area-id> [<auth-type>] [stub-area|sa [<cost>] [sum]]
```

add Specifies to add an OSPF area.

<area-id> Specifies the area ID. Specify the area ID in dotted decimal notation (xxx.xxx.xxx.xxx, where each “x” is an integer from 0 – 9). The area ID must be unique within the Autonomous System.

NOTE The area ID 0.0.0.0 is reserved for the Autonomous System’s backbone and is already present.

<auth-type> Specifies the authentication type. Specify one of the following:

none|n Specifies that the OSPF area being added does not use authentication.

simple-password|sp Specifies that a password is required for OSPF packets sent within this area.

md5|m Specifies that MD5 authentication is required for OSPF packets sent within this area. See RFC 1321 for information about MD5 authentication.

The ASN-9000 default is none (no authentication).

When an OSPF interface is added to this area (using the **`interface`** command), specify the actual simple password or MD5 authentication key ID.

NOTE All OSPF routers in an area must have the same authentication type and string. Also, all OSPF routers on a particular network should use the same authentication string.

<stub-area-cost|sac <cost> Specifies that the area is a stub area. Configuring an area as a stub area reduces OSPF overhead in the network by reducing the amount of OSPF route information flooded to the OSPF routers in the stub area.

The OSPF software does not flood external routing information (information about other Autonomous Systems) into the stub area. Internal routers in the stub area reach Autonomous Systems by using the default route to the stub area's Area Border router.

The OSPF software advertises the default route automatically. Note that a stub area's default route is unrelated to the default routes you can define in the `ip` subsystem. OSPF uses the default routes it defines in preference to manually configured default routes.

The cost is the metric for the default route out of the stub area. The stub area's Area Border router advertises the cost as part of the default route. You can specify a value from **`1`** through **`65535`**. The default is **`1`**.

[sum] Sum parameter set to **Yes** will produce the following: Import summary LSAs. Sum parameter set to **No** produces the following: Don't import summary LSAs

Following is an example of the OSPF **area add** command specifying the authentication type as "simple password":

```
88:ASN-9000:ip/ospf# area add 147.128.136.39 sp
Added Area 147.128.136.39 , Authentication Type "simple password"
```

5.13.2.6.1 Deleting an OSPF Area

To delete an OSPF area, first disable OSPF routing then issue the following command:

`area delete|del` ***`<area-id>`*** **`|all`**

delete|del Deletes an OSPF area from the ASN-9000.

<area-id>|all Specifies the area to delete. To delete all OSPF areas defined on this ASN-9000, specify **`all`**.

The example below illustrates the sequence of these commands:

```
90:ASN-9000:ip/ospf# ospf disable
OSPF Routing: Disabled
91:ASN-9000:ip/ospf# area del 147.128.136.39
Deleted Area 169.144.86.64
```

5.13.2.7 Displaying an OSPF Area

The **`area|ar [show|sh] [`***`<area-id>`***`]`** command is used to display information about the OSPF areas configured on the ASN-9000.

Following are some examples of the information displayed by this command. In the following example, information is displayed for all the OSPF areas configured on the ASN-9000.

```
Area Id         Auth    Import ASEs #SPFRun     #ABR       #ASBR        #LSA     SumCost
147.128.136.39 sp      Yes (Type 5)1           0          0           0       Yes
OSPF Area Count: 1
```

The fields in this display show the following information:

Area ID Displays the OSPF area ID assigned using the **`area add`** command. The area ID is a 32-bit integer expressed in dotted decimal notation. The area ID 0.0.0.0 is the backbone area ID and is added automatically by the ASN-9000.

Auth Displays the authentication type assigned for this area using the **area add** command. The authentication type can have one of the following values:

noNo authentication is required for this area.

spA simple password is required for this area.

md5MD5 authentication is required in this area. See RFC 1321 for information about MD5.

Import ASEs Specifies whether this area is configured to import external LSAs from other Autonomous Systems. Shows "yes" or "no." A "no" in this column indicates that an area is a stub area.

#SPFRun Indicates the number of times the ASN-9000 has calculated this area's intra-area route table. This number is reset to zero if OSPF routing is disabled, the software has been rebooted, or the ASN-9000 has been powered down.

# ABR Indicates the number of Area Border routers that can be reached from this area.

# AS BR Indicates the number of Autonomous System Border routers that can be reached from this area.

#LSA Indicates the number of LSAs in this area's LSA database. This number does not include external LSAs.

Sum "Yes" specifies to import summary LSAs; "no" specifies not to import external LSAs.

Cost If this area is a stub area, the metric for the stub area is indicated in this field. If this area is not a stub area, this field contains dashes (-----). A stub area's metric can be assigned when adding the area using the **area add** command.

5.13.2.8 Area Set/Unset Command

The area set|unset command sets or unsets the area cost for the specified stub area. To set or unset the area cost, issue the following command:

```
area|ar set <area-id> <stub-area-cost|sac <cost>>
```

<area-id> A 32-bit integer in the form "XXX.XXX.XXX.XXX".

<cost> Cost for the stub area.

5.13.2.9 Adding an OSPF Interface to an Area

An OSPF interface is not automatically added to the ASN-9000 when an IP interface is added. The OSPF interface should have the same address as the IP interface. Before adding an OSPF interface, OSPF must be enabled, and an area -id and interface has to be added in the ip/ospf subsystem.

1. Enable OSPFusing the following command:

```
[ospf] enable|disable
```

```
3:ASN-9000:ip/ospf# enable
```

2. Add an area-id, using the following command:

```
area|ar add <area-id> [<auth-type>] [stub-area|sa [cost] [sum]]
```

```
4:ASN-9000:ip/ospf# ar add 12.23.5.3
Added Area 12.43.5.3 , Authentication Type "none"
```

3. Once the interface has been added in the ip subsystem, add an OSPF interface using the following command:

```
interface|it add <ip-addr> area|ar <area-id>  [auth <key-str>|-k
        <keyid>] [cost|c <Cost>] [priority|p <priority>]
```

<ip-addr> Specifies the IP address of the interface. Specify the interface in dotted decimal notation (xxx.xxx.xxx.xxx, where each “x” is an integer from 0 – 9)

<area-id> Specifies the OSPF area in which the OSPF interface is being placed. An OSPF interface can belong to only one area. The area must already be configured (using the **area add** command.

[auth <key-str>] Specifies the authentication string. For a simple password, specify any combination of up to eight numbers, letters, and special characters. If the area to which this interface is being added does not require an authentication string, use empty quotation marks ("").

[auth <keyid>] Specifies the value to be used as the MD5 Authentication Key that is defined in the nvram subsystem. If a string shorter than 16 octets is supplied, the string is left-justified and padded to 16 octets, on the right, with nulls (0x00).

[cost|c <cost>] Specifies the cost (number of hops) using this interface.

[priority|p <priority>] Specifies the priority used in electing the Designated Router. A value of 0 denotes that the Router is ineligible to become DR.

[xdelay|x <trans-delay>] Specifies I/F transmission delay.

[rint|r <rxmt-int>] Specifies LSA retransmission interval.

[hint|h <hello-int>] Specifies time between hello packets.

[rdint|d <rtr-dead-int>] Specifies time to declare a Router Dead, usually a multiple of the hello interval.

[pint|pi <poll-int>] Specifies time between hello(s) after an nbma router is assumed dead.

The command below adds an interface:

```
10:ASN-9000:ip/ospf# it add 169.144.86.54 ar 12.43.5.3
Added OSPF Interface 169.144.86.54 to Area 12.43.5.3
```

4. To show the OSPF interface, issue the following command:

```
interface|it [show] <ip-addr>
```

Entering **interface** from the ip/ospf subsystem displays the following information:

```
11:ASN-9000:ip/ospf# it
IP Address     Area Id        DR             Backup DR      Type    Admin  Oper
--------------------------------------------- -------------------- -----  ----
169.144.86.54  12.43.5.3      169.144.86.54  0.0.0.0        Bcast   Enl    Enl

OSPF Interface Count: 1
```

where

IP Address The IP address of this OSPF interface.

Area ID A 32-bit integer uniquely identifying the area to which this interface connects. Area ID “0.0.0.0” is the OSPF backbone.

DR The IP Address of the Designated Router.

Backup DR The IP address of the Backup Designated Router.

Type The type of this OSPF interface. This could be one of Bcast, NBMA or PToP.

Admin The administrative status of this interface.

Oper The operational status of this interface.

OSPF Interface Count Displays the number of OSPF interfaces displayed in the above table.

NOTE Type-of-Service (TOS) cannot be specified. The ASN-9000 uses TOS 0 (zero, the IP TOS).

5.13.2.10 Enabling the ASN-9000 Switch as a System Border Router

The ASN-9000 can be enabled to function as an Autonomous System Border router. A switch enabled to be an Autonomous System Border router automatically exports OSPF routes to the networks outside of the OSPF Autonomous System and imports routes from the networks outside the Autonomous System. To enable or disable the ASN-9000 as an Autonomous System Border router, issue the following command:

```
asbd enable|disable
```

enable|disable Specifies whether to enable or disable the ASN-9000 to function as an Autonomous System Border router. If **enable** is specified, the ASN-9000 can exchange route information between RIP and OSPF. If **disable** is specified, the software cannot exchange route information. The default is **disable**.

To view the changes you’ve made, issue the following command:

```
asbd [show]
```

5.13.2.11 Interface Command

In most Autonomous Systems, the ASN-9000 defaults for the OSPF interface parameters are appropriate for the Autonomous System. However, if change to a specific interface parameter is required, use the following command to do so:

```
interface|it set <ip-addr> [area|ar <area-id>]
[auth <key-str>|-k <keyid>][cost|c <cost>]
[priority|p <priority>][xdelay|x <transdelay>]
[rint|r <rxmt-int>][hint|h <hello-int>][rdint|d <rtr-dead-int>]
[pint|pi <poll-int>]
```

<ip-addr> Specifies the IP address of the interface. Specify the interface in dotted decimal notation (xxx.xxx.xxx.xxx, where each “x” is an integer from 0 – 9).

The IP address must already be present in the IP interface table before it can be used to create an OSPF interface.

[ar <area-id>] Specifies the OSPF area in which the OSPF interface is being placed. An OSPF interface can belong to only one area. The area must already be configured (using the **area add** command.

[auth <key-str>] Specifies the authentication string. For a simple password, specify any combination of up to eight numbers, letters, and special characters. If the area to which this interface is being added does not require an authentication string, use empty quotation marks ("").

<keyid> The md5 authentication keyid must be in the range 0-255

[cost|c <cost>] Specifies the cost of using this interface. The ASN-9000 advertises the cost in Router Links Advertisements. Specify a cost from **1** through **32**. This parameter does not have a default value. The cost depends upon the wire speed of the segment on which the interface is being added. Unless the cost needs to be changed, FORE Systems recommends that this argument be omitted and use the value determined by the ASN-9000.

[priority|p <priority>] Specifies this interface's priority during the election process for the Designated Router (DR). The interface with the highest priority number is elected as the DR. The interface with the second-highest priority number is elected as the Backup Designated Router (BDR).

Specify a priority from 0 through 255. Priority increases from **1** (lowest) to **255** (highest). A priority of **0** (zero) makes this interface ineligible for becoming the DR. The default is **1**.

If all OSPF interfaces within an Autonomous System have the same priority, the DR and BDR are elected based on the interface addresses. The interface with the highest OSPF address is elected as the DR. The interface with the second-highest OSPF address is elected as the BDR.

NOTE

Generally, an OSPF router has only one interface per area. If the ASN-9000 has multiple interfaces to the same area, the interface priority still applies.

[xdelay|x <transdelay>] Specifies the interface transmission delay, which is the estimated number of seconds it takes to transmit a Link State Update packet over this interface. The ASN-9000 adds the transmission delay specified to the ages of the LSAs contained in the Link State Update packets sent on this interface.

Specify a delay from **1** through **3600**. The default is **1**. Refer to RFC 2178 for information about choosing transmission delay.

[rint|r <rxmt-int>] Specifies the retransmission interval. The retransmission interval is the number of seconds between transmissions of LSAs to the OSPF routers adjacent to this interface. The retransmission interval also is used when transmitting Database Description and Link State Request packets.

Specify an interval from **1** through **3600**. The default is **5**.

[hint|h <hello-int>] Specifies the hello interval. The hello interval is the number of seconds between transmission of Hello packets on this interface. Specify an interval from **1** through **65536**. The default is **10**.

NOTE

The hello interval (**hint**) and the router-dead interval (**rdint**) must match on neighbors. That is, the values for these parameters must match the values on the neighbor for these parameters. If the OSPF neighbor also is a ASN-9000 system, ensure that the values match by accepting the defaults for these parameters. If the neighbor is not a ASN-9000, the value on the neighbor or on the ASN-9000 may need to be changed so that the values on both routers match.

[rdint|d <rtr-dead-int>] Specifies the router-dead interval. The router-dead interval is the number of seconds the OSPF neighbors should wait before declaring that the ASN-9000, as an OSPF router, is down.

Specify a router-dead interval from **1** through **65536**. Specify an interval that is an even multiple of the Hello interval. The default is **40**.

<poll-int> Specifies the time between hello(s) after an NBMA router is assumed dead.

5.13.2.12 Adding Network Ranges

It is not necessary to add network ranges to OSPF areas. The ASN-9000 automatically advertises all the networks on all the OSPF interfaces on the switch to other OSPF routers. Network ranges can be added to reduce OSPF overhead or to hide certain networks from other OSPF routers.When a network range is added to an area, link-state information for the networks within the range is summarized in the LSAs sent by the switch to its OSPF neighbors. Therefore, if there are many networks within an area, adding the networks as a network range can help reduce OSPF overhead.

In addition, the **noadv** argument can be used with the **net-range** command to prevent the switch from advertising routes to the networks within a network range. When the switch sends LSAs to its neighbors, LSAs for the networks in the hidden network range are not sent to the switch's neighbors. Therefore, other routers in the Autonomous System do not learn about the hidden networks.

NOTE None of the networks within the network range added to an area can be in other areas.

To add a network range to an OSPF area, issue the following command:

net-range add *<area-id> <net> <mask>* [noadv|na]

<area-id> Specifies the OSPF area. The area must already have been added to the switch. To add an area, use the **area add** command.

<net> Specifies an IP network address in dotted decimal notation (xxx.xxx.xxx.xxx, where each “x” is an integer from 0 – 9).

<mask> Specifies the IP mask associated with the IP network address specified for the <*net*> argument. The mask indicates the portion of the IP network address that is to be regarded as the network portion of the address. Specify the mask in dotted decimal notation (ex: 255.255.255.0).

noadv|na Prohibits the OSPF software from advertising this network range in the LSAs transmitted by the switch to its OSPF neighbors. If this argument is used, other OSPF routers do not learn about the presence of the network range.

In the following example, the network range specified by IP address 200.200.200.0 and subnet mask 255.255.255.0 is added to area 12.43.5.3. When area 12.43.5.3 sends LSAs to other areas, the LSAs contain summary information for the networks within the network range, instead of detailed link-state information for each network within the network range.

```
5:ASN-9000:ip/ospf# nr add 12.43.5.3 200.200.200.0 255.255.255.0
Added net-range [area :12.43.5.3 net 200.200.200.0 mask 255.255.255.0]
```

If the **noadv** argument had been specified with the command, the area would not report the networks within the specified network range.

5.13.2.13 Deleting Network Ranges

To delete a network range, issue the following command:

net-range delete|del *<area-id> <net> <mask>*

<area-id> Specifies the OSPF area.

<net> Specifies the IP network address.

<mask> Specifies the subnet mask associated with the IP address.

Here is an example of this command.

```
8:ASN-9000:ip/ospf# nr del 12.43.5.3 200.200.200.0 255.255.255.0
Deleted net-range [area :12.43.5.3 net 200.200.200.0 mask 255.255.255.0]
```

After a network range space has been deleted, the ASN-9000 sends detailed link-state information for each network, instead of summarizing the link-state information for the entire range.

5.13.2.14 Displaying Network Ranges

Use the **net-range|nr [show] [<area-id>]**command to display information about the network ranges assigned to the areas configured on the ASN-9000. If the optional <*area-id*> argument is omitted, summary information is displayed for all the network ranges in all the areas. To display network-range information for a specific area, use the <*area-id*> argument.

Here is an example of the information displayed by the **net-range show** command. In this example, the optional<*area-id*> argument is omitted. Only one network range is listed in the display, indicating that only one OSPF network range has been configured.

```
12:ASN-9000:ip/ospf# nr
Area ID            Net             Mask              Advertise
---------------    --------------- ---------------   ---------
12.43.5.3          200.200.200.0   255.255.255.0     Yes
```

The fields in this display show the following information:

Area ID The OSPF area that contains the network range.

Net The IP address of the network or subnet portion of the network range. The network number is ANDed with the subnet mask (see the Mask field) to make the network range.

Mask The subnet mask that is ANDed with the network number (see the Net field) to make the network range.

Advertise Indicates whether this network range is advertised to other areas. The advertise state can be Enabled or Disabled. The advertise state is enabled by default. To prevent from advertising the network range to other areas, use the **`noadv`** argument with the **`net-range`** command.

5.13.2.15 Displaying OSPF Neighbors

The **`neighbor show`** command is used to display information about OSPF neighbors. Here is an example of the information displayed by this command.

```
114:ASN-9000:ip/ospf# neighbor show
Interface       Nbr IP Address  Router ID       Pri State    Events RTrQ  Type
OSPF Neighbor Count: 0
```

The fields in this display show the following information:

Nbr IP Address Neighbor's interface IP address.

Router ID The ID of the OSPF router that contains the neighbor.

Pri The priority of the OSPF router that contains the neighboring interface. The priority is used when the ASN-9000 elects a DR and a BDR. If the priority is 0 (zero), the OSPF router is ineligible to become the DR or BDR.

State The state of the relationship with the neighboring interface's router. The state can be one of the following:

down:The switch has not received recent information from the neighbor.

attempt:The switch has not received recent information from the neighbor, but the software is attempting to contact the neighbor by sending Hello packets. The Hello interval can be changed using the **`hint`** argument of the **`nset`** command.

init:The switch recently received a Hello packet from the neighbor.

two Way:Communication between the switch and the neighbor now is bi-directional.

ex star:tThe switch and its neighbor are beginning to exchange their link-state databases.

exchange:The switch is sending its link-state database to the neighbor.

loading:The switch is sending Link State Request packets to the neighbor, requesting the LSAs that are more recent than the information contained in the link-state database the switch sent to that neighbor. The switch updates its link-state database with the new LSAs received from the neighbor.

full:The switch and the neighbor have finished exchanging their link-state databases.

For more information about these states, refer to RFC 2178.

Events The number of times the state of the neighbor relationship (see the State field) has changed. Refer to RFC 2178.

RTrQ The current length of the retransmission queue.

type The type of this OSPF interface. This could be one of Bcast, NBMA or PToP.

5.13.3 Trace Settings, Trace Level, and Trace Class Commands

The trace settings, trace level, and trace class commands are used for debugging purposes only.

5.13.3.1 Trace Settings

The **`tracesettings|tr`** command displays the current trace settings. Trace settings include trace levels and enabled trace classes. By default, all area, non-interface and interface trace settings are displayed. The syntax to display the current trace settings is:

```
tracesettings|tr [show] [<ifaddr> | area|ar <area-id>]
```

where:

<ifaddr> Show the tracesettings set on the specified interface <ifaddr> or all OSPF interfaces if <ifaddr> is not specified.

area|ar Show the tracesettings set on the specified area id <area-id>.

Entering **`tracesettings`** from the ip/ospf subsystem displays:

```
6:ASN-9000:ip/ospf# tr

                    -- Trace settings --

Entity         Action       Level        Enabled classes
--------------- -------     ----------   --------------------------
OSPF:
               Print        (warning)    gen timer route txpkt rxpkt
Area: 12.43.5.3
               Print         (warning)    gen timer route txpkt rxpkt
OSPF Intf: 169.144.86.54
               Print        (warning)    gen timer route txpkt rxpkt
```

5.13.3.2 Trace Level

The **tracelevel** command sets trace levels of info, notice, or warning on a specified OSPF interface address (<ifaddr>), area (<area-id>) or all areas and interfaces if no interface or area is specified. Entering **tracelevel** with no parameters displays the current trace level settings. The syntax for this command is:

```
tracelevel|trl [n]set level-name [<ifaddr> | area|ar <area-id>]
```

The example below set the trace lever to warning on all configured interfaces:

```
31:ASN-9000:ip/ospf# trl set warning
```

Entering **tracelevel** from the `ip/ospf` subsystem displays:

```
32:ASN-9000:ip/ospf# trl

                    -- Trace settings --

Entity          Action    Level         Enabled classes
--------------- ------- ---------- --------------------------
OSPF:
                Print   (warning)   gen timer route txpkt rxpkt
Area: 12.43.5.3
                Print   (warning)   gen timer route txpkt rxpkt
OSPF Intf: 169.144.86.54
                Print   (warning)   gen timer route txpkt rxpkt
```

5.13.3.3 Trace Class

The **traceclass** command, enables or disables trace class settings of rxpkt, txpkt, route, or gen on a specified OSPF interface address (<ifaddr>), OSPF area (<area-id>), or on all interface addresses and areas if no interface or area is specified. Entering **traceclass** with no parameters displays the current trace level settings that are configured. The syntax for this command is:

```
traceclass|trc [n]enable class-name [<ifaddr> | area|ar <area-id>]
traceclass|trc [n]disable class-name [<ifaddr> | area|ar <area-id>]
```

The command below disables the route class on all configured OSPF interfaces:

```
33:bASN-9000:ip/ospf# trc disable route
```

Entering **`traceclass`** from the `ip/ospf` subsystem displays:

```
34:ASN-9000:ip/ospf# trc

                    -- Trace settings --

Entity          Action    Level         Enabled classes
--------------- ------- ---------- -------------------------
OSPF:
                Print   (warning)   gen timer txpkt rxpkt
Area: 12.43.5.3
                Print   (warning)   gen timer txpkt rxpkt
OSPF Intf: 169.144.86.54
                Print   (warning)   gen timer txpkt rxpkt
```

5.13.3.4 Displaying OSPF Link-State Advertisements

Use the following command to display information about a link-state database:

```
lsdb [show][area|ar <area-id>][lstype<type>]
        [lsid<lsdb-id>][rid<router-id>]
```

<lsdbid> Specifies the ID of a specific LSA.

<rid> Specifies the OSPF router ID of the router from which the link-state database was received.

<type> Specifies the LSA type, which can be one of the following types:

r : Router LSA

n : Network LSA

s : Summary LSA

a : Autonomous System Summary LSA

e : External LSA

If the optional arguments are omitted, summary information is displayed for all the LSAs present in the LSA database. To display detailed information about a specific LSA, use the optional arguments.

Here are some examples of the information displayed by this command. In the first example, summary information for all LSAs in the switch’s LSA database is displayed.

```
16:ASN-9000:ip/ospf# lsdb show
Area Id   LSA Type      Link State ID   Router ID   Sequence
--------- -------------  --------------  ----------  -----------
0.0.0.0   routerLink     1.1.1.1         1.1.1.1     -2147483552
0.0.0.0   routerLink     2.2.2.2         2.2.2.2     -2147483303
0.0.0.0   routerLink     3.3.3.3         3.3.3.3     -2147483615
0.0.0.0   routerLink     5.5.5.5         5.5.5.5     -2147483576
0.0.0.0   networkLink    80.100.1.3      3.3.3.3     -2147483635
0.0.0.0   networkLink    129.213.72.2    5.5.5.5     -2147483635
0.0.0.0   summaryLink    87.0.0.0        2.2.2.2     -2147483348
0.0.0.0   summaryLink    150.1.100.0     1.1.1.1     -2147483578
0.0.0.0   summaryLink    150.1.100.0     3.3.3.3     -2147483632
1.1.1.1   routerLink     3.3.3.3         3.3.3.3     -2147483635
1.1.1.1   networkLink    150.1.100.3     3.3.3.3     -2147483646
1.1.1.1   summaryLink    44.0.0.0        3.3.3.3     -2147483640
1.1.1.1   summaryLink    80.100.0.0      3.3.3.3     -2147483640
1.1.1.1   summaryLink    80.200.0.0      3.3.3.3     -2147483640
<example truncated for brevity>
```

The fields in this display show the following information:

Area ID The OSPF area from which the LSA was received.

Lsdb Type The type of LSA.

Link State ID The ID of the LSA, in dotted-decimal notation. The LSA ID is determined by the type of the LSA, as described in Table 5.1:

Table 5.1 - LSA Type to LSA ID

LSA Type	LSA ID
An Internal router's LSA (routerLink).	The originating router's OSPF router ID.
A network LSA (networkLink).	The IP interface address of the network's DR (Designated Router).
A summary LSA (summaryLink).	The destination network's IP address.
An Autonomous System Border router's LSA (asSummaryLink).	The OSPF router ID of the Autonomous System Boundary router described by the LSA.
An Autonomous System Border router's external LSA (asExternalLink).	The destination network's IP address.

Route ID The OSPF router from which the LSA was received.

Sequence The sequence number of the LSA. The sequence number is a 32-bit signed integer. A higher sequence number indicates a more recent LSA. Use the LSA sequence numbers to detect old or duplicate LSAs.

In the following example, detailed information is displayed about a specific LSA.

```
17:ASN-9000:ip/ospf# lsdb show 1.1.1.1 1.1.1.1 r 0.0.0.0
Detailed View
Area ID                     : 0.0.0.0
Link State Database Type    : routerLink
Link State ID               : 1.1.1.1
Originating Router ID       : 1.1.1.1
Sequence Number             : -2147483552
Advertisement Age           : 1503
Advertisement Checksum      : ccac
The OSPF Link State Database Advertisement: (26 per line)
00 00 02 01 01 01 01 01 01 01 01 01 80 00 00 60 cc ac 00 30 03 00 00 02 81 d5
48 02 81 d5 48 01 02 00 00 0a 03 03 03 03 96 01 64 01 04 00 00 0a
```

The fields in this display show the following information:

Area ID The OSPF area from which the LSA was received.

Link State Database Type The type of LSA. The LSA can be one of the following types:

`routerLink`Internal router LSA

`networkLink`Network LSA \

`summaryLink`Summary LSA

`asSummaryLink`Autonomous System Border router LSA

`asExternalLink`External LSA

Link State ID Area ID The ID of the LSA. The LSA ID depends upon the type of the LSA as defined in Table 5.1.

Originating Router ID Area ID The OSPF router from which the LSA was received.

Sequence Number Area The sequence number of the LSA. The sequence number is a 32-bit signed integer. A higher sequence number indicates a more recent LSA. Use the LSA sequence numbers to detect old or duplicate LSAs.

Advertisement Age Area ID The age, in seconds, of the LSA.

Advertisement Checksum Area The checksum for the LSA.

The OSPF Link State Database Advertisement Area The contents of the LSA, in hexadecimal.

5.13.3.5 External LSDB

The external link state database command, **external-lsdb**, command displays link state database information on external link state advertisements (LSAs). Specifying the link state database id and the router id shows detailed information for the LSA. The syntax of this command is:

```
external-lsdb|elsdb [show] [<lsdb-id> <router-id]
```

Entering **external-lsdb** from the ip / ospf subsystem displays the following:

```
33:ASN-9000:ip/ospf# elsdb
LSA Type   Link State ID   Router ID       Sequence
---------- -------------- -------------- --------
OSPF External LSA Count: 0
34:ASN-9000:ip/ospf#
```

where

LSA Type Type of external LSA.

Link State ID Link State Database ID in the form “XXX.XXX.XXX.XXX”.

Router ID Originating AS Router ID in the form “XXX.XXX.XXX.XXX”.

Sequence Sequence number of the LSA.

OSPF External LSA Count Displays a count of external LSA’s.

5.13.3.6 Enabling the Return-Code Prompt

The return-code prompt is intended primarily for automated interactions with the ASN-9000 command-line interface. To enable printing of command return codes in the next UI prompt, issue the following command:

```
rcprompt enable
```

To disable the return-code prompt, issue the following command:

```
rcprompt disable
```

5.13.3.7 Adding a Virtual-Link

Depending upon how the OSPF network is configured, it is possible for some areas to be completely disconnected from one another. Areas become disconnected from one another when they are not attached to the backbone and do not share a Border router.

The ASN-9000 can automatically link disconnected areas using the automatic virtual-link feature. This feature links together ASN-9000s configured as OSPF routers when they are separated from one another.

If some of the OSPF routers in your Autonomous System are not ASN-9000s, areas that are separated can be linked by defining a virtual link between the areas. The virtual link makes the disconnected areas virtual neighbors. LSAs from an area reach that area's virtual neighbor by travelling through a transit area. The transit area is an area between the two virtual neighbors that passes traffic between the neighbors.

NOTE

The transit area must be added to the OSPF network before configuring the virtual link.

To add a virtual link, use the following command:

```
virtual-link|vlink add <area-id> <router-id> [auth <key-str>|-k
      <keyid>][xdelay|x <trans-dly>] [rint|r <rxmt-int>]
         [hint|h <hello-int>]   [rdint|d <rtr-dead>]
```

The values and defaults for these arguments are the same as the arguments and defaults for the **nset** command. The example below adds a virtual link:

```
2:ASN-9000:ip/ospf# vlink add 1.1.1.1 147.128.9.7
```

5.13.3.8 Deleting a Virtual-Link

To delete a virtual link, issue the following command:

```
virtual-link|vlink delete|del <area-id> <router-id>
```

<router-id> Specifies the OSPF Router ID of the virtual neighbor. Specify the router ID in dotted decimal notation (xxx.xxx.xxx.xxx, where each “x” is an integer from 0 – 9).

NOTE

Use the **virtual-link del** command to delete a virtual link created by the software automatically using the automatic virtual-link feature. However, if the automatic virtual-link feature is enabled, the software adds the link again. To prevent the software from adding a virtual link again, disable the automatic virtual-link feature by issuing the **auto-vlink disable** command.

5.13.3.9 Displaying Virtual-Links

Use the following command to display information about a virtual link:

```
virtual-link|vlink [show] [<area-id> <router-id>]
```

If the optional arguments are omitted, summary information is displayed for all the virtual links that exist between this and other OSPF routers. To display detailed information about a virtual link, use the optional arguments.

Following is an example of the information displayed by this command. In this example, summary information is displayed. The switch in this example has only one virtual link to another OSPF router.

```
130:ASN-9000 :ip/ospf# vlink show
Area ID        Neighbor ID    IP Address     If State       Nbr State
1.1.1.1        3.3.3.3        147.138.9.7    up             full
```

The fields in this display show the following information:

Area ID The OSPF area on the local side of the virtual link.

Neighbor ID The OSPF router ID of this neighbor.

IP Address The IP address of the router on the remote end of the Virtual Link. Routers can have many IP addresses. This IP address is the one assigned to the remote router’s segment that connects the remote router to the ASN-9000.

IF State The state of the virtual interface. The state can be one of the following:

`up`The interface can be used to send and receive OSPF route information.

`down`The interface is unavailable for sending or receiving OSPF traffic. The interface’s link state is be reported as down in LSAs sent from this OSPF router.

Nbr State The state of the virtual interface. The state can be one of the following:

`down`The switch has not received recent information from the neighbor.

`attempt`The switch has not received recent information from the neighbor, but the software is attempting to contact the neighbor by sending Hello packets. The Hello interval can be changed by using the **`hint`** argument of the **`nset`** command.

`init`The switch recently received a Hello packet from the neighbor.

`two Way`Communication between the switch and the neighbor now is bi-directional.

`ex start`The switch and its neighbor are beginning to exchange their link-state databases.

`exchange`The switch is sending its link-state database to the neighbor.

`loading`The switch is sending Link State Request packets to the neighbor, requesting the LSAs that are more recent than the information contained in the link-state database the switch sent to that neighbor. The switch updates its link-state database with the new LSAs received from the neighbor.

`full`The switch and the neighbor have finished exchanging their link-state databases.

In the following example, detailed information is displayed for a specific virtual link.

```
21:ASN-9000:ip/ospf# virtual-link show 1.1.1.1 3.3.3.3
Area ID                             : 1.1.1.1
Router ID                           : 3.3.3.3
IP Address                          : 150.1.100.3
Transmit Delay                       : 1
Retransmission Interval             : 5
Hello Interval                      : 10
Router Dead Interval                : 60
Authorization Key String            :
Authorization Failures              : 0
Virtual Interface State             : up
Virtual Interface Events            : 1
Virtual Neighbor State              : full
Virtual Neighbor Events             : 5
Virtual Neighbor Retransmission Que : 0
```

The fields in this display show the following information:

Area ID The OSPF area on the local side of the virtual link.

Router ID The router ID of the OSPF router on the local end of the virtual link. (The ASN-9000 OSPF router ID.)

Neighbor IP Address The IP address of the router on the remote end of the Virtual Link. Routers can have many IP addresses. This IP address is the one assigned to the remote router’s segment that connects the remote router to the ASN-9000.

Transit Delay The interface transmission delay for this interface.

Retransmission Interval The retransmission interval for this interface.

Hello Interval The Hello interval for this interface.

Router Dead Interval The Hello interval for this interface.

Authentication Key String The authentication string for the interface. The authorization string is specified by the <*key-str*> argument of the **`interface`** command. If this field is blank, then no authentication string is required for this interface.

Authentication Failures The number of times another OSPF router tried to use this interface but did not supply the correct authorization string.

Virtual Interface State The state of the virtual interface. The state can be one of the following:

`up`The interface can be used to send and receive OSPF route information.

`down`The interface is unavailable for sending or receiving OSPF route information. The interface's link state is reported as down in LSAs sent from this OSPF router.

Virtual Interface Events The number of times the state (see the Virtual Interface State field) has changed since OSPF routing was enabled.

Virtual Neighbor State The state of the relationship with the OSPF router on the remote end of the virtual link. The state can be one of the following:

`down`The switch has not received recent information from the neighbor.

`attempt`The switch has not received recent information from the neighbor, but the software is attempting to contact the neighbor by sending Hello packets. The Hello interval can be changed by using the **`hint`** argument of the **`nset`** command.

`init`The switch recently received a Hello packet from the neighbor.

`two Way`Communication between the switch and the neighbor now is bi-directional.

`ex start`The switch and its neighbor are beginning to exchange their link-state databases.

`exchange`The switch is sending its link-state database to the neighbor.

`loading`The switch is sending Link State Request packets to the neighbor, requesting the LSAs that are more recent than the information contained in the link-state database the switch sent to that neighbor. The switch updates its link-state database with the new LSAs received from the neighbor.

`full`The switch and the neighbor have finished exchanging their link-state databases.

Virtual Neighbor Events The number of times the relationship with the remote end of the virtual link has changed since OSPF routing was enabled. The state is displayed in the `Virtual Neighbor State` field.

5.13.3.10 Timed Commands

In some router implementations, packet processing can affect timer execution. Multiple routers are attached to a single network, all doing broadcasts, can lead to the synchronization of routing packets (which should be avoided). If timers cannot be implemented to avoid drift, small random amounts should be added to/subtracted from the timer interval at each firing.

5.13.3.11 Statistics Command

As soon as OSPF forwarding is enabled, the ASN-9000 begins collecting OSPF statistics. The **`stats show`** command is used to display statistics or the **`stats clear`** command to clear statistics.

5.13.3.12 Displaying OSPF Statistics

To display the OSPF statistics, issue the following command:

```
stats show
```

Here is an example of the information displayed by the **`stats`** command:

```
62:ASN-9000:ip/ospf# stats
OSPF Packet Statistics (since last stats clear):


Good Hello Rx                    : 0
Good DB Description Rx           : 0
Good Link-State Req Rx           : 0
Good Link-State Update Rx        : 0
Good Link-State Ack Rx           : 0
Good Hello Tx                    : 124
Good DB Description Tx           : 0
Good Link-State Req Tx           : 0
Good Link-State Update Tx        : 0
Good Link-State Ack Tx           : 0
LSA Information:
    External LSA Count       :0
    External LSA Checksum Sum:0x00000000
    OriginateNewLSAs         :2
    RxNewLsas                :0
Memory Stats Since Last Reboot (in bytes) :
LSA header partition :
    Size = 48000         Used = 0
    Available = 48000    Peak Use = 0%
Database description list partition:
    Size = 123840         Used = 0
    Available = 123840    Peak Use = 0%
LSDB list partition:
    Size = 60000         Used = 0
    Available = 60000    Peak Use = 0%
Database retransmission partition:
    Size = 200000        Used = 0
    Available = 200000   Peak Use = 0%
LSA Ack partition:
    Size = 120000        Used = 0
    Available = 120000   Peak Use = 0%
```

5.13.3.13 Clearing OSPF Statistics

To clear OSPF statistics, issue the following command:

```
stats clear
```

Here is an example of this command.

```
137:ASN-9000:ip/ospf# stats clear
```

The ASN-9000 clears the counters for the statistics and begins collecting statistics again. Statistics also are cleared if OSPF routing is disabled, the software is rebooted, or the ASN-9000 is powered down.

CHAPTER 6 Internetwork Packet Exchange (IPX)

This chapter describes the commands in the `ipx` subsystem and describes how to use the commands to configure and manage the ASN-9000 as an IPX router. The commands in this subsystem are used to perform the following tasks:

- Allocate memory for IPX routing
- Display the IPX configuration
- Add, show, and delete IPX interfaces
- Enable IPX routing
- Display, add, and delete IPX routes
- Display or clear the IPX route cache
- Configure IPX RIP
- Add, display, and delete IPX servers
- Configure IPX helper addresses
- Display and clear IPX statistics
- Customize the IPX routing behavior

6.1 Accessing the IPX Subsystem

To access the ipx subsystem, enter the following command from any runtime command prompt:

ipx

Listed below are the commands and subsystems available at this level:

```
2:ASN-9000:ipx# ?

ipx subsystem:

cache                                >rip
config                                ripsap-ctrl|rsct
diag_ipx                              route|rt
getmem                               >sap
helper                                server
interface|it                          stats
ipx| [ipx] enable | [ipx] disable     type20-forwarding|t20fw
large-rip-sap-pkt|lpkt                t20stats
one-rip-entry|onere                   type20-port-forwarding|tpfw
```

6.2 Allocating Memory for IPX Routing

Before the `ipx` subsystem can be used, memory must be allocated. Regardless of how much main memory is available, memory must be specifically allocated for use by the `ipx` subsystem.

FORE Systems recommends that memory for the **`ipx`** subsystem be allocated immediately after booting to ensure that the memory requested is available.

To allocate memory for the `ipx` subsystem, issue the following command:

```
getmem
```

6.3 Displaying the IPX Configuration

The current IPX settings can be displayed by issuing the **`config show`** command. The example below shows the display produced by this command:

```
3:ASN-9000:ipx# config
IPX Configuration:
IPX Router:                   Memory Available
IPX Forwarding:               enabled
IPX Type20 Packet Forwarding: disabled
IPX Helper Feature          : disabled
Large RIP and SAP Packets:    disabled
One Rip 'Equal' Entry:        disabled
RIP broadcast timer interval: 60
SAP broadcast timer interval: 60
RIP aging timer interval:    180
SAP aging timer interval:    180
```

where

IPX Router Indicates whether main memory has been allocated for the IPX subsystem.

IPX Forwarding Indicates whether IPX forwarding is enabled or disabled. The default setting is disabled.

IPX Type20 Packet Forwarding Indicates that the ASN-9000 is configured to forward type-20 IPX packets. The default setting is enabled.

IPX Helper Feature Indicates the setting of the IPX helper feature. When enabled, this feature allows the ASN-9000 to forward unknown IPX broadcast packets.

Large RIP and SAP Packets Indicates whether the ASN-9000 isenabled to forward large (greater than 576 bytes) IPX RIP and SAP packets. The default setting is disabled.

One-rip-entry Indicates whether the configuration is set to accept first 'equal' RIP route to network .

RIP broadcast timer interval Indicates how often RIP broadcasts are sent. Default is 60 seconds.

SAP broadcast timer interval Indicates how often SAP broadcasts are sent. Default is 60 seconds.

RIP aging timer interval Indicates how many seconds a learned, unused IPX route can remain in the route table before it is removed by the aging mechanism. Default is 180 seconds. If a value other than the default is selected, the RIP aging timer interval is always three times the RIP-packet aging interval.

SAP aging timer interval Indicates how many seconds a learned, unused IPX server can remain in the server table before it is removed by the aging mechanism. The default is 180 seconds. If a value other than the default is selected, the SAP aging timer intervals is always three times the SAP-packet aging interval.

Any of the IPX configuration items listed in this display can be configured. This chapter describes the commands used to set these items.

6.4 Adding IPX Interfaces

The **interface add** command is used to assign an IPX interface (sometimes referred to as a network number) to one or more segments. When adding an interface, an entry is made in the route table to show that the network is directly connected to the specified segment. (See Section 6.7.1.) After you have allocated memory, using the **getmem** command, follow the sequence of commands as outlined below to add an IPX interface.

1. Before you can add an IPX interface, you must enable IPX forwarding.The syntax for this command is:

```
ipx enable <port>
```

The example below enables IPX forwarding:

```
2:ASN-9000:ipx# ipx enable
IPX forwarding enabled.
```

2. Add an IPX interface using the following command syntax:

```
interface|it add <segmentlist> <network>
    [mtu <mtu>] [met[ric] <metric>]
     [encap enet|802.3|802.2|snap]
```

where

<segmentlist> Specifies the segment number(s) assigned to the IPX interface. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments.

NOTE If more than one segment number per interface is specified, an IPX interface for a VLAN is created. Refer to Chapter 5 for information on configuring VLANs.

<network> Specifies an IPX network number. Specify a hexadecimal number in the range from **1** through **fffffffe**.

[mtu <mtu>] Specifies the maximum transmission unit (number of octets) for packets forwarded on this segment. Specify a number in the range from **576** through **1500**. The default is **576**.

[met[ric <metric>] Specifies an additional cost (extra hops) of using the interface. Specify a cost in the range 1–14. When the ASN-9000 reports this subnet using RIP, it adds the additional cost to the reported metric. The default metric is 0.

The following examples show the use of this command.

```
81:ASN-9000:ipx# it add 2.1 1001
Segment 2.1, Network 0x1001, MTU 576, Cost 0,
Added
```

The first command creates an IPX interface on segment 2.1. Because this interface is intended to be used as the primary route to the ASN-9000 from a router, no cost is specified.

Here is an example of the **interface add** command used to add an IPX network to more than one ASN-9000 segment. This command creates an IPX VLAN.

```
105:ASN-9000:ipx# it add 2.6-2.8 2012 mtu 1500 encap snap
Segment 2.6, Network 0x2012, MTU 1500, Cost 0, Frame type SNAP
Added
Segment 2.7, Network 0x2012, MTU 1500, Cost 0, Frame type SNAP
Added
Segment 2.8, Network 0x2012, MTU 1500, Cost 0, Frame type SNAP

Added
```

6.4.1 Deleting IPX Interfaces

The **interface delete** command is used to delete an IPX interface. The syntax for this command is:

interface|it del[ete] ***<segmentlist>*****|all** ***<network>*****|all**

where

<segmentlist>|all Specifies the segment(s) to delete. If **all** is specified, the network number is removed from all segments.

<network>|all Specifies the IPX network to delete. If **all** is specified, all IPX networks are deleted from the specified segment(s).

The command below deletes the IPX interface from segment 2.6 :

```
10:ASN-9000:ipx# it del 2.6 2012
Segment 2.6: network 2012 deleted
```

6.5 Displaying IPX Interfaces

Network numbers assigned to segments can be viewed by using the **`interface show`** command. The syntax for this command is:

`interface|it [show]` ***`<segmentlist> <network>`***

where

<segmentlist> Specifies the segments to display IPX interface information. If a list or range of segments is specified, information is shown for only those segments that have IPX interfaces.

<network> Specifies the IPX network for which to display information.

The display includes the segment state—UP, if the segment is up, or DOWN, if the segment is disabled or if the automatic segment-state detection mechanism has determined the segment to be down. Following is an example of the information displayed by this command.

```
15:ASN-9000:ipx# it
  Segment    Network Address  MTU    Encapsulation     State      Cost
  -------     --------------- ---    -------------      -----     ----
  2.1        00001001         576    802.3             UP         0
  2.7        00002012         1500   802.2/SNAP        DOWN       0
  2.8        00002012         1500   802.2/SNAP        DOWN       0
```

6.6 Enabling IPX Routing

Enable IPX forwarding after defining the IPX interfaces (see Section 6.4). By enabling IPX forwarding, the IPX software can send and receive RIP and SAP updates, and respond to RIP and SAP requests from stations. Use the following command to enable IPX forwarding:

```
[ipx] enable|disable
```

where

enable|disable Specifies whether IPX forwarding is to be enabled or disabled. The default state is disabled.

6.6.1 Adding and Deleting IPX Routes

Use the **`route add`** command to assign the route to be used when forwarding to a particular network. The syntax for this command is:

```
route|rt add <network> <gw-net> <gw-addr> s[eg[ment]] <seg> h[ops]
                   <hops> t[icks] <ticks>
```

where

<network> Specifies the destination IPX network number, a 32-bit value expressed as up to eight hexadecimal digits. Specify a number in the range from **`1`** through **`fffffffe`**.

<gw-net> Specifies the network number of the gateway (IPX router) through which packets for the destination network are to be routed. This network number must be one of the network numbers that is already configured on the segment specified by the <*seg*> argument. Specify a number in the range from **`1`** through **`fffffffe`**.

<gw-addr> Specifies the IPX node number of the gateway (router) to which packets for the destination network should be forwarded. An IPX node number is actually a 48-bit MAC-layer address. Such an address is expressed as six hexadecimal bytes separated by hyphens.

The gateway should be a device connected to a network that is directly attached to the segment specified in the <*seg*> argument.

<seg> Specifies the segment on which a packet should be forwarded to reach the specified gateway and, eventually, the specified network.

<hops> Specifies the number of hops to the destination, that is, how many gateways a packet must go through to reach the specified network.

A hop-count of **1** corresponds to a direct connection. (Note, however, that you cannot add a route to a network that is directly attached.)

The maximum number of hops is **`15`**; a hop-count of **`16`** is synonymous with “infinity” and means that the specified network is unreachable.

<ticks> Specifies the typical delay expected for a packet to reach its destination, measured in 55-mS “ticks.”

In Ethernet, FDDI, and other networks with bandwidths greater than 1 Mb/s, each network is assumed to create a delay of one tick. If a route includes only such networks, the number of ticks should be set equal to the number of network segments in the route, which is the number of hops plus 1. However, routing paths that include slow, wide-area links (ex: 56 Kb/s leased lines) should have a larger number of ticks to account for the slow links.

Ticks are represented in IPX by 16-bit integers, so the practical maximum number of ticks is far less than the number that can be entered here. A statically-entered IPX route is always marked as “UP” when added. The route is automatically marked as “DOWN” when the corresponding segment is disabled, either manually in the `bridge` subsystem or automatically by the automatic segment-state detection mechanism.

When routing a packet to a remote network, the IPX routing software selects the route with the lowest number of ticks, regardless of whether it is a static or dynamic route. When two or more routes to a remote network have an equal number of ticks, the route with the smallest number of hops is chosen. An example of the **`route add`** command is shown below:

```
7:ASN-9000:ipx#rt add 008ffff9 96aabb69 0-0-99-88-88-8 2.32 3
Route to 008ffff9 via 96aabb69: added.
```

The result of this command is that packets directed to network 008ffff9 are forwarded on segment 2.3 to a gateway with address 0-0-99-88-88-88, and can expect to require a total of 2 hops and 3 ticks to reach a station on the destination network.

6.6.2 Deleting IPX Routes

Static routes can be completely eliminated using the **route del** command. The syntax for this command is:

route|rt del[ete] ***<network> <gw-net> <gw-addr>***

where

<network> Specifies the destination IPX network number, a 32-bit value expressed as up to eight hexadecimal digits.

<gw-net> Specifies the network number of the gateway (IPX router).

<gw-addr> Specifies the IPX node number of the gateway (router).

```
7:ASN-9000:ipx#rt del 008ffff9 96aabb69 0-0-99-88-88-8
Route to 008ffff9 via 96aabb69: deleted.
```

6.6.3 Displaying IPX Routes

The **route show** command is used to display the IPX route table. The syntax for this command is:

route|rt [show] [-c|-r|-t] [<disprestrictors>]

where

-c | -r | -t Restricts the display to one of the following:

-c Only directly connected entries

-r Only remotely attached entries

-t Displays the total count of UP and DOWN routes.

<seglist> Specifies the segment(s) to display route information.

[<disp-restrictors>] Configures the display to show restrictors: [[s[eg[ment][s]]]=]<seglist> n[et[work]]=<network>

Here is an example of the display produced by this command:

```
60:ASN-9000:ipx# route show

      Destnet   Gway-net  Gway-nodeaddr      Hops  Ticks  State  Age Sgmts
      --------  --------  ----------------   ----  -----  -----  --- -----
      00001001  --------  ------------         1     2     UP    ---   1
      00002002  --------  ------------         1     2     UP    ---   6
      55ccdd55  --------  ------------         1     2     UP    ---   1
      55ccdd55  --------  ------------         1     2     UP    ---   2
      55ccdd55  --------  ------------         1     2     UP    ---   3
      008ffff9  96aabb69  00-00-99-88-88-88    2     3     UP    ---   8
      054ffff9  f4f4f4f4  00-00-99-22-22-22    2     3     UP    ---   4
      064ffff9  f4f4f4f4  00-00-99-22-22-22    2     4     UP    ---   4
      011ffff9  96aabb69  00-00-99-11-11-11    2     3     UP    ---   3
      165ffff9  00fabcab  00-00-99-44-44-44    2     3     UP    ---   9

Total no. of routes = 10 (10 UP, 0 DOWN)
```

This command displays the following information about IPX routes:

where

Destnet IPX network number of the destination network.

Gway-net If the destination is not directly attached, this field contains the IPX network number of the gateway (IPX router) through which packets for the destination are to be routed.

Gway-nodeaddr If the destination is not directly attached, this field contains the node address of the IPX gateway (router) through which packets for the destination are to be routed.

Hops The number of gateways, including the ASN-9000, that a packet must go through to reach the destination. If a network is directly attached, the hop-count is 1.

Ticks The number of 55-mS ticks that can be expected for a packet to reach its destination. If all of the network segments along the route have a bandwidth of 1 Mb/s or more, the number of ticks generally equals the number of hops plus 1. Otherwise, it is larger to account for the slower segments.

State The state of the route; UP or DOWN. When a segment goes down, its state is updated in the interface table. All routes that use this segment are marked DOWN in the route table, and all servers that are not accessible except through this segment are marked as DOWN in the server table.

When the segment comes back up, its state is again updated in the interface table. All routes that use this segment are marked as UP in the route table, and servers that are now accessible through this segment are marked as UP in the server table.

Age For dynamic routes, the number of seconds that have elapsed since this routing information was received. The Age field displays “---” for direct/static routes. For RIP entries, the Age field displays how long it has been since a routing update for the route has been received.

Ports Lists the segments on which packets for this destination should be forwarded.

The software does not contain a command to directly take a static route DOWN. To take DOWN a static route, use the **`route delete`** command to remove the route.

6.7 Displaying and Clearing the IPX Route Cache

The IPX route cache shows, for each segment, the most recently used destination networks. At any time, an at-a-glance picture of IPX routing activity in the network can be displayed by displaying the IPX route cache.

6.7.1 Displaying the Route Cache

The **cache show** command is used to display the IPX route cache. The syntax for this command is:

```
cache [show] [<disprestrictions>]
```

where

[<disprestrictions>] Sets the display to show restrictors:
[[s[eg[ment][s]]]=]<seglist>

Here is an example of the output produced by this command. The cache displayed in this example is for an ASN-9000 containing 14 segments.

```
66:ASN-9000:ipx# cache show
IPX router cache:
Segment 1.1: empty
Segment 1.2: empty
Segment 1.3: 011ffff9, 96aabb69
Segment 1.4: f4f4f4f4, 054ffff9, 064ffff9
Segment 1.5: empty
Segment 1.6: empty
Segment 2.1: empty
Segment 2.2: 00000022
Segment 2.3: 00fabcab, 165ffff9
Segment 2.4: empty
Segment 2.5: empty
Segment 2.6: empty
Segment 3.1: empty
Segment 3. 2 : empty
```

NOTE

The contents of the route cache can change quite rapidly. As a result, successive **cache show** commands can give different results.

6.7.2 Clearing the Route Cache

The **cache clear** command removes all entries from all segments in the route cache. After the cache is flushed, new entries are again added, using the cache's "most-recently-used" algorithm. Thus, the **cache clear** command can be used to ensure that all entries displayed by a subsequent **cache show** command are fresh. In the following example, the route cache is flushed once and then quickly displayed two times:

```
67:ASN-9000:ipx# cache clear
IPX router cache flushed
68:ASN-9000:ipx# cache show
IPX router cache:
Segment 1.1: empty
Segment 1.2: empty
Segment 1.3: 96aabb69
Segment 1.4: f4f4f4f9
Segment 1.5: empty
Segment 1.6: empty
Segment 2.1: empty
Segment 2.2: empty
Segment 2.3: empty
Segment 2.4: empty
Segment 2.5: empty
Segment 2.6: empty
Segment 3.1: 00001fd1
Segment 3.2: empty
69:ASN-9000:ipx# cache show
IPX router cache:
Segment 1.1: empty
Segment 1.2: empty
Segment 1.3: 96aabb69, 011ffff
Segment 1.4: f4f4f4f4, 054ffff
Segment 1.5: empty
Segment 1.6: empty
Segment 2.1: empty
Segment 2.2: 00000022
Segment 2.3: 00fabcab
Segment 2.4: empty
Segment 2.5: empty
Segment 2.6: empty
Segment 3.1: 00001fd1
Segment  3. 2 : empty
```

6.8 Configuring IPX RIP and SAP Parameters

Earlier sections in this chapter described how to add static entries to the IPX RIP and SAP tables. However, there are additional RIP and SAP options that can be configured:

- Generating updates on a per-segment or per-VLAN basis.
- Generating large (greater than 576 bytes) IPX RIP and SAP packets.
- Talk and listen (send and receive) settings for each interface or segment.

6.8.1 Setting the Control Type

The RIP and SAP control type can be set to change the RIP and SAP update mechanism. Using the **set ripsap-ctrl** command, the ASN-9000 can be configured to generate and send a copy of each RIP and SAP packet on a per-VLAN basis instead of on a per-segment basis.

If the IPX configuration does not contain IPX VLANs, performance can be the same whether configured to generate updates on a per-segment basis or on a per-VLAN basis. In this case, it is recommended that the configuration be left in its default state: generate updates on a per-segment basis.

However, if the configuration does include IPX VLANs, performance can be enhanced by configuring the per-VLAN method for generating RIP and SAP updates. When the control type is changed to **vlan**, less time is spent generating RIP and SAP updates, because only a single update is generated for each network, even if the network spans multiple segments. To change the RIP and SAP update method, issue the following command:

```
ripsap-ctrl|rsct set [normal|n vlan|v]
```

where

normal|n Specifies that RIP and SAP updates are generated on a per-segment basis. This is the default.

vlan|v Specifies that RIP and SAP updates are generated on a per-VLAN basis.

If no parameter is used with this command, the current control type is displayed.

NOTE

This command affects only IPX RIP and SAP updates. It has no effect on IP RIP updates.

The example below sets the RIP and SAP control type to normal:

```
5:ASN-9000:ipx# rsct set n
```

6.8.1.1 Displaying the RIP and SAP Control Type

To display the RIP / SAP control type, issue the following command:

`ripsap-ctrl|rsct [show]`

The command produces the following results:

```
399:ASN-9000:ipx# ripsap-ctrl show
ripsap-ctrl-type:      normal
```

6.8.1.2 Adjusting the Interval and Aging Timers

The RIP and SAP timers can be adjusted. The ASN-9000 IPX implementation generates and transmits RIP and SAP updates at regular intervals. RIP updates contain information about known IPX routes. SAP updates contain information about known IPX servers.

The default interval for RIP and SAP updates is 60 seconds. Every 60 seconds, IPX RIP and SAP updates are generated and transmitted. Depending on whether RIP and SAP updates are configured to use the per-segment method or the network method, updates are generated for each segment or for each network.

Aging is a mechanism that periodically clears learned entries from the RIP and SAP tables. At a specified interval (the aging interval) the ASN-9000 determines which of the learned entries in the table have not recently been used. For proper RIP and SAP reporting, the aging interval must be at least three times the duration of the broadcast interval. If an entry is not used during the specified interval, it is discarded. A separate broadcast interval and aging timer are maintained for IPX RIP and for IPX SAP. To set interval and aging timers for RIP, issue the following command:

`timers set` ***`<transmit-intvl>`*** **`[`*`<rip-age>`*`]`**

where

<transmit-intvl> Sets the RIP broadcast interval. Specify a value from **`60`** to **`600`** seconds. The default is **`60`** seconds.

<rip-age> Sets the RIP age timer. If specified, the RIP age timer value must be at least three times the value of the RIP broadcast interval. Specify a value between **`201`** and **`1800`** seconds. If unspecified, this argument defaults to three times the value of the RIP broadcast interval.

Following is an example of this command:

```
82:ASN-9000:ipx/rip# timers set 100 300
```

To set interval and aging timers for SAP, issue the following command:

`timers set` ***`<transmit-interval-time>`*** **`[`*`<aging-time>`*`]`**

where

<transmit-interval-time> Sets the SAP broadcast interval. Specify a value from **`60`** to **`600`** seconds. The default is **`60`** seconds.

<aging-time> Sets the SAP age timer. If specified, the SAP age timer value must be at least three times the value of the SAP broadcast interval. Specify a value between **`201`** and **`1800`** seconds. If unspecified, this argument defaults to three times the value of the SAP broadcast interval.

Following is an example of this command:

```
86:ASN-9000:ipx/sap# timers set 100 300
```

6.8.2 Setting Talk and Listen for RIP and SAP

The following sections describe the commands available in the ipx/rip subsystem for setting and disabling talk and listen parameters for RIP and SAP.

The IPX software advertises and receives IPX routing information using the IPX RIP .IPX server information is advertised and received using the SAP.

NOTE The RIP protocol used by IPX is different from RIP used in IP.

The commands for displaying the talk and listen (send and receive) settings for IPX RIP and SAP differ depending upon the update method used:

- If the update method is per-segment, use the **`penable`** command in both the `ipx/rip` and `ipx/sap` subsystems within the `ipx` subsystem.

If the update method is per-VLAN, use the **`nenable`** command in both the `ipx/rip` and `ipx/sap` subsystems within the `ipx` subsystem.

6.8.2.1 Setting RIP Parameters

To enable IPX RIP sending (**`talk`**) or receiving (**`listen`**), use the **`talk penable`** and **`listen penable`** commands in the `ipx/rip` subsystem. The syntax for these commands is:

```
talk|ta penable <seglist>|all
talk|ta nenable <network>
listen|li penable <seglist>|all
listen|li nenable <network>
```

where

<seglist>|all Specifies the segments to enable IPX RIP sending and/or receiving. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments. If **`all`** is specified, IPX RIP is enabled for all segments.

<network> Specifies the following:

talk|ta Enables the sending of RIP update packets to the specified network.

listen|li Enables the learning of routes from RIP packets received from the specified network.

The examples below enable RIP sending and receiving on segments 1.2 through 1.14:

```
37:ASN-9000:ipx/rip# ta penable 1.2-1.14
38:ASN-9000:ipx/rip# li penable 1.2-1.14
```

6.8.2.2 Disabling RIP Parameters

To disable IPX RIP **`talk`** or **`listen`**, use the **`talk pdisable`** and **`listen pdisable`** commands in the `ipx/rip` subsystem. The syntax for these commands is:

```
talk|ta pdisable <seglist>|all
talk|ta ndisable <network>
listen|li pdisable <seglist>|all
listen|li ndisable <network>
```

The example below disables RIP receiving on network 2012:

```
48:ASN-9000:ipx# rip li ndisable 2012
```

6.8.2.3 Setting SAP Parameters

To enable IPX SAP sending (**talk**) or receiving (**listen**), use the **talk penable** and **listen penable** commands in the ipx/sap subsystem. The syntax for these commands is:

```
talk|ta penable <seglist>|all
talk|ta nenable <network>
listen|li penable <seglist>|all
listen|li nenable <network>
```

where

<seglist>|all Specifies the segments for which IPX SAP sending or receiving is set. Specify a single segment, a comma-separated list of segments, or a hyphen-separated range of segments. If **all** is specified, IPX SAP is enabled for all segments.

<network> Specifies the following:

talk|ta Enables the sending of SAP update packets to the specified network.

listen|li Enables the learning of routes from SAP packets received from the specified network.

The commands below enable sending and receiving of SAP update packets on network 1001:

```
50:ASN-9000:ipx/sap# ta nenable 1001
51:ASN-9000:ipx/sap# li nenable 1001
```

6.8.2.4 Disabling SAP Parameters

To disable IPX SAP **talk** or **listen**, use the **talk pdisable** and **listen pdisable** commands in the ipx/sap subsystem. The syntax for these commands is:

```
talk|ta pdisable <seglist>|all
talk|ta ndisable <network>
listen|li pdisable <seglist>|all
listen|li ndisable <network>
```

The command below disables sending if SAP update packets on network 1001:

```
52:ASN-9000:ipx/sap# ta ndisable 1001
```

6.8.3 Displaying the Configuration

To display the talk and listen (send and receive) settings for RIP and SAP updates, use the **config show** command in both the `ipx/rip` and `ipx/sap` subsystems. The syntax for this command is:

`config show [`*`<seglist>`*`] [`*`<network>`*`]`

where

<seglist> Specifies the segments to display IPX RIP and IPX SAP configurations. If no segment is specified, all RIP and SAP control table entries are displayed.

<network> Network address of the network to display RIP and SAP control table entries. If no network is specified, all RIP and SAP control table entries are displayed.

Following is an example of the display produced by this command if **`vlan`** was selected in the **`set ripsap-ctrl`** command:

```
53:ASN-9000:ipx/sap# config all
Warning: current rip control mode is vlan; ignoring segment restrictors
SAP VLAN Configuration:
Network      Talk     Listen
-------      ----     ------
00001001     no       yes
00002012     no       no

55:ASN-9000:ipx/rip# config all
Warning: current rip control mode is vlan; ignoring segment restrictors
RIP VLAN Configuration:
Network      Talk     Listen
-----------   ----    ------
00001001     no       no
00002012     no       no
```

If **`normal`** was selected in the **`set ripsap-ctrl`** command, entering the **`config show`** command from the `ipx/rip` subsystem will produce a per-segment configuration display :

```
91:ASN-9000:ipx/rip# config 1.1-1.6

Segment       Talk      Listen
-------       ----      ------
 1.1             yes       yes
 1.2             yes       yes
 1.3             yes       yes
 1.4             yes       yes
 1.5             yes       yes
 1.6             yes       yes
```

6.8.4 Equal RIP Route

To enable or disable accepting the first equal RIP route to the network, issue the following command from the ipx subsystem:

```
one-rip-entry|onere enable|disable
```

6.9 Using the Server Table

Information about NetWare file servers and other NetWare services are stored in a data structure called the server table. The IPX routing software maintains a server table containing information that it uses when advertising services and responding to server information requests using SAP. The table contains two types of servers:

Dynamic servers Learned through the SAP. IPX file servers, print servers, and other service providers advertise their existence using SAP. This information is learned by all IPX routers in the network. When an IPX station requires a service, it uses SAP to request server information from the nearest router.

Static servers Configured by a system administrator, using the **`server add`** command. The IPX routing software always has SAP enabled, and services are always being discovered and advertised dynamically. Although the information learned through SAP is usually sufficient for good network behavior, there might be occasions in which permanent entries are desired in the server table. For example, permanent entries can be made in the server table to ensure quick availability of service information after a network outage. Static service assignments can be used for this purpose.

NOTE Before adding a server to the IPX server table, a route (to the IPX route table) must be added to the server’s net.

When responding to IPX stations' requests for the information on the "nearest" server of a given type, the server with the best route, as determined from the route table, is selected regardless of whether the server is static (added to the server table permanently by the **`server add`** command) or dynamic (learned through SAP). If there are equally good routes to two or more servers, the server with the least number of hops in the server table is chosen.

6.9.1 Displaying the Server Table

To display known IPX servers, issue the following command:

```
server [show] [-f|-a|-t] [<disprestrictors>]
```

where

[-f|-a|-t] Specifies the type of entries to display:

- -f Displays the entire server name, up to 48 characters. Otherwise, a maximum of 24 characters is displayed to keep the display within an 80-character line.
- -t Displays only the total count of UP and DOWN server entries.
- -a Displays the network number and MAC address of the next-hop gateway.

[<disprestrictions>] Sets the display to show restrictors:
[[s[eg[ment][s]]]=]<seglist>

Table 6.1 - Service Types

Mnemonics	Server-type(hex)
PRINT-QUEUE	0003
FILE-SERVER	0004
JOB-SERVER	0005
PRINT-SERVR	0007
ARCHIVE-SVR	0009
REM-BRIDGE	0024
ADVRT-PRINT	0047

Here is an example of the output produced by this command.

```
19:ASN-9000:ipx# server
Server-typeSrvr-net  Server-node    Sock  Hop   State Sgmt   Age     Server-name
------------------- ------------   ----- ------ ----- ---- ---      ---------------
030c        00001001  08-00-09-6f-16-a2 400c 2   UP    2.1     31      0800096F16A283CG
030c         00001001 08-00-09-1c-0c-3 400c2    UP    2.1      20     0800091C0C3383C2

Total no. of servers = 2 (2 UP, 0 DN)
```

This command displays the following information from the server table:

where

Server-type Specifies the type of service, either a mnemonic or a 16-bit number in the range **0** through **fffe**, expressed as up to four hexadecimal digits.

Srvr-net The IPX network number of the server.

Server-node The IPX node number of the server.

Sock The IPX socket number on which the server accepts requests for service.

Hop The number of gateways, including the ASN-9000, that a packet must go through to reach the server. If the server is on a directly-attached network, the hop-count is 1.

State This is the state of the server; possible states are “UP” and “DOWN.”

Segment The segment on which the entry was learned.

Age For dynamic servers, the number of seconds that have elapsed since this information was received.

Server-name The name of the server, up to 48 ASCII characters.

6.9.2 Adding a Static Server

To add a server to the server table, use the **server add** command. Here is the syntax for this command.

```
server add <s-type> <s-net> <s-addr>
```

where

<s-type> Specifies the type of service, either a mnemonic or a 16-bit number in the range **0** through **fffe**, expressed as up to four hexadecimal digits (see Table 6.1)

<s-net> Specifies the IPX network on which the server resides, a 32-bit number expressed as up to eight hexadecimal digits.

NOTE

The **server add** command is not accepted if there is no known route to the server's network at the time the command is given. Specify a number in the range from **1** through **ffffffe**.

Here is an example of the **server add** command:

```
3:ASN-9000:ipx# server add 4 fabcab 0-0-88-88-88-88 1010 2 phsrvr
Server phsrvr of type 0004 on net 00fabcab: added.
```

6.9.3 Deleting a Static Server

A static server assignment can be deleted by using the **server delete** command. The syntax for this command is :

```
server del[ete] <s-type>
```

where

<s-type> Specifies the type of service, either a mnemonic or a 16-bit number in the range **0** through **fffe**, expressed as up to four hexadecimal digits (see Table 6.1)

Here is an example of the use of this command.

```
72:ASN-9000:ipx# server del 4
Server eng-server of type 0004: deleted from table.
```

6.10 Using IPX Helper

This section describes how to use the IPX Helper feature. IPX Helper lets the ASN-9000 forward unknown IPX broadcast packets, which normally would be dropped, to specified networks. This feature forwards unknown IPX broadcast packets without using the IPX SAP protocol. When an IPX helper address is assigned to a segment, and an unknown IPX broadcast packet with the specified destination socket number is received on that segment, the following responses result:

- The IPX broadcast packet destination network number and destination node address are replaced with the number and address specified in the **helper add** command.
- The IPX broadcast packet is then forwarded onto all other segments.

To use IPX Helper, it must first be enabled by issuing the following command:

```
helper enable|disable
```

6.10.1 Adding an IPX Helper Address

The **helper add** command is used to add an IPX helper address to a segment. Here is the syntax for this command:

```
helper add <seglist> <network> <node address> s[ock[et]] <socket>
```

<seglist> Specifies a segment, a comma-separated list of segments, or a hyphen-separated range of segments.

<network> Specifies a network number or the value **ffffffff** to specify all net broadcast.

<node address> Specifies the unicast address or the broadcast address **ff-ff-ff-ff-ff**.

s[ock[et]] <socket> Specifies the destination socket number.

Here is an example of how to add an IPX Helper address. In this example, a broadcast address is defined:

```
95:ASN-9000:ipx# helper add aabbccdd ff-ff-ff-ff-ff-ff ffff
```

6.10.2 Displaying an IPX Helper

The **helper show** command is used to display IPX helper addresses assigned for all segments. Here is an example of the information produced by this command:

```
220:ASN-9000:ipx# helper show

SEGMENT   NETWORK    NODE ADDRESS         SOCKET NUMBER
-------   -------    ------------         -------------
   1      aabbccdd   ff-ff-ff-ff-ff-ff    ffff
```

6.10.3 Deleting an IPX Helper Address

The **helper delete** command is used to delete an IPX helper address assigned to a segment. The syntax for this command is:

helper del[ete] ***<seglist>***

<seglist> Specifies a segment, a comma-separated list of segments, or a hyphen-separated range of segments.

6.11 Showing and Clearing Statistics

The **stats** command is used to display IPX or type-20 packet statistics. Here is the syntax for this command:

stats [show] [-t]

-t Optionally displays statistics collected since the most recent switch reset, rather than those collected since the most recent clear.

Here is an example of the output produced by the **stats** command.

```
80:ASN-9000:ipx# stats
IPX statistics: count since last stats clear
Datagrams received:                 2302091
Header errors received:             0
Address errors received:            0
Datagrams forwarded:                2302091
Unknown Broadcast packets forwarded: 0
Unknown protocols received:         0
Incoming datagrams discarded:       0
Datagrams delivered to higher layer: 2258
Datagrams sent:                     6658
```

Here is an example of the output produced by the **stats type20** command.

```
81:ASN-9000:ipx# t20stats
Type-20 statistics: count since last stats clear
Packets   received:                 0
Packets   forwarded:                0
Packets   discarded:                0
Packets   in error:                 0
```

Here is an example of the use of the **-t** argument with the **stats** command. In this example, IPX statistics collected since the last switch reset are displayed.

```
83:ASN-9000:ipx# stats -t
IPX statistics: Total count since last system reset
Datagrams received:                 2305309
Header errors received:             0
Address errors received:            0
Datagrams forwarded:                2305309
Unknown Broadcast packets forwarded: 0
Unknown protocols received:         0
Incoming datagrams discarded:       0
Datagrams delivered to higher layer: 2261
Datagrams sent:                     6664
```

To clear statistics, use the **stats clear** command.

6.12 Customizing the IPX Configuration

To enable or disable the forwarding of type-20 packets for the entire switch, issue the following command:

```
type20-forwarding|t20fw enable|disable
```

The command below enables type-20 forwarding:

```
60:ASN-9000:ipx# t20fw enable
Type20 forwarding enabled.
```

6.12.1 Type-20 Forwarding for Segments

The **`type20-port-forwarding`** command is used to show whether type-20 packet forwarding is enabled or disabled on specific segments. The syntax for this command is:

`type20-port-forwarding|tpfw penable|pdisable <seglist>`

penable|pdisable Specifies whether type-20 packet forwarding is to be enabled or disabled. The default is **`type20-port-forwarding`** enabled.

<seglist> Specifies a segment, a comma-separated list of segments, or a hyphen-separated range of segments for which to enable or disable type-20 packet forwarding.

The command below enables type-20 forwarding on segment 2.1:

```
61:ASN-9000:ipx# tpfw penable 2.1
Okay
```

6.12.2 Enabling Large Packets

In software version 3.0, IPX RIP and IPX SAP packets larger than 576 bytes (the default minimum) can be generated. To change the default, use the **`enable large-rip-sap-pkt`** command to enable the software to generate large RIP and SAP packets. The syntax for this command is:

`large-rip-sap-pkt|lpkt enable|disable`

enable|disable Specifies that the ASN-9000 generate or not generate IPX RIP or IPX SAP packets larger than 576 bytes. The default for **`large-rip-sap-pkt`** isdisabled.

NOTE The MTU setting for the IPX interfaces defined on the switch needs to be more than 576 bytes to generate larger RIP and SAP packets.

The following command enables large RIP and SAP packets to be generated:

```
63:ASN-9000:ipx# lpkt enable
Large RIP & SAP packet generation enabled
```

6.13 Configuring IPX Translation Bridging

IPX translation bridging allows one or more IPX networks that span across FDDI and segments to be configured using different packet encapsulations. Without altering the configurations of individual devices, IPX translation bridging enables Ethernet devices with different encapsulation types to communicate with each other. This feature is especially useful if the IPX network consists largely of Ethernet devices using 802.3 encapsulation, the default encapsulation type in Novell IPX software versions 2.2 through 3.11. However, if the network name is not in the IBT table, IPX translation bridging does not occur, and normal bridging does. This section describes how to perform the following tasks:

- Show the switch's IPX translation-bridging configuration
- Add, show, and delete IPX translation-bridging interfaces

NOTE

IPX translation bridging is independent of IPX routing—they are mutually exclusive. It is recommended that both IPX translation bridging and IPX routing not be enabled. However, if both IPX translation bridging and IPX routing are enabled, IPX routing takes precedence over IPX translation bridging.

6.13.1 Encapsulation Types

When IPX translation bridging is used, the Ethernet encapsulation types to be used on each IPX network are specified. For each IPX network number, the Ethernet encapsulations to be used on that network can be specified. Table 6.2 lists the combinations of encapsulation types you can specify.

Table 6.2 - Encapsulation Types

	ENET	802.2	802.3	SNAP
Ethernet	4	4	4	4

6.13.2 Configuration Requirements

Although IPX translation bridging is simple to configure, the following conditions must be met:

- The servers attached to the segments in an IPX translation bridging network must be configured to have the same network number as the “IPX translation-bridging” network number configured on the ASN-9000. If a server’s network number cannot be changed to correspond to the IPX translation-bridging network defined on the switch, change the network number to match the server.
- Servers and clients must be configured to have the same encapsulation type as the type specified for the appropriate medium in the IPX translation-bridging network. For example, a client attached to an Ethernet segment must be configured to use the same Ethernet encapsulation type as the one defined for the corresponding IPX translation-bridging network. However, if encapsulation types on the server or client cannot be changed, the encapsulation types of the client or server can be configured on the switch.

6.13.2.1 Enabling IPX Translation Bridging

Before the IPX translation-bridging feature can be used, IPX translation bridging must be enabled. To enable IPX translation bridging, issue the following command from the `bridge` subsystem:

```
ipx-br-translation|ibt enable|disable
```

enable|disable Specifies whether IPX translation bridging is to be enabled or disabled.

Here is an example of the use of this command:

```
1:ASN-9000:bridge# ibt enable
IPX translation bridging is now enabled
```

6.13.2.2 Adding IPX Translation-Bridging Interfaces

To create an IPX translation-bridging network, use the following command:

```
ipx-br-translation|ibt add <network> <ether-encap> <fddi-encap>
```

<network> Specifies the IPX network number to apply the encapsulation settings.

<ether-encap> Specifies the encapsulation type to be used for Ethernet packets. Packets bridged from FDDI to this network number are converted to this encapsulation. Specify one of the following:

enet	Ethernet Type II encapsulation.
802.3	Raw 802.3 encapsulation.
802.2	802.3 with an LLC header.
snap	802.3 with LLC and SNAP headers.

NOTE The default Ethernet encapsulation type used in Novell IPX versions 2.2 through 3.11 is 802.3. The default for versions 3.12 through 4.x is 802.2.

<fddi-encap> Specifies the encapsulation type to be used for packets translated to FDDI. Specify one of the following:

802.3	Raw 802.3 encapsulation.
802.2	802.3 with an LLC header.
snap	802.3 with LLC and SNAP headers.

NOTE The default FDDI encapsulation type used in Novell IPX versions 2.2 through 3.11 is 802.3, the same type used for Ethernet devices. Similarly, the default FDDI encapsulation type used in versions 3.12 through 4.X is 802.2.

Here are some examples of how to use this command. In these examples, definitions are created for IPX translation-bridging networks 100, 200, and 300:

```
24:ASN-9000:bridge# ibt add 100 802.2 snap
IPX network 100 added to the translation table
25:ASN-9000:bridge# ibt add 200 802.2 802.2
IPX network 200 added to the translation table
26:ASN-9000:bridge# ibt add 300 802.3 snap
IPX network 300 added to the translation table
```

6.13.2.3 Displaying IPX Translation-Bridging Interfaces

Definitions for the IPX translation-bridging networks defined on the ASN-9000 can be displayed at any time. To display the definitions, use the following command:

ipx-br-translation|ibt [show] [*<network>*]|[-t]

<network> Specifies an IPX translation-bridging network number.

-t Displays only the total number of entries in the IPX translation-bridge table.

Here are some examples of displays produced by this command. In the first example, no specific network number is given, so all individual entries are displayed, as well as the total number of entries. In the second example (prompt 7), the **-t** argument is used to display the total number of IPX translation-bridging entries.

```
29:ASN-9000:bridge# ibt
IPX Translation Bridging: Enabled
IPX Network    Ethernet Encap FDDI Encap
----------     -------------- ----------
100            802.2          802.2/SNAP
200            802.2
300            802.3          802.2/SNAP
Total entries:  3

30:ASN-9000:bridge# ibt -t
IPX Translation Bridging: Enabled
IPX Network    Ethernet Encap FDDI Encap
-----------    -------------- ----------
total entries: 3
```

6.13.2.4 Deleting IPX Translation-Bridging Interfaces

To delete the encapsulation settings assigned to a network number, use the following command:

ipx-br-translation|ibt del *<network>*|all

<network>|all Specifies an IPX translation-bridging network number. If **all** is specified, all IPX translation-bridging networks are deleted.

Here is an example of the use of this command:

```
8:ASN-9000:bridge# ibt del all
All IPX networks deleted from the IPX translation table
```

APPENDIX A Well-Known Ports

This appendix lists the well-known names provided in RFC 1340 that the ASN-9000 supports. When configuring an IP or TCP filter, either the port number or the well-known name can be supplied to specify the destination port of packets to either block or accept. Supply the port number or well-known name in the <*dstseg*> field of templates for any TCP and IP filters being created. The <*dstseg*> field is used with the following TCP and IP filter commands:

- tcp tcp-filter add
- ip ip-fil-acs-ctrl add

When an IP packet comes in on an Ethernet segment, the Ethernet header is stripped away. The packet then relies on the IP header to begin routing it through the LAN to its eventual destination. In the IP header, the protocol type field denotes the kind of packet that follows, such as ARP, TCP, or UDP.

If the protocol type field indicates a TCP or UDP packet, then that packet is travelling from a source port to a destination port; a 16-bit number represents each port. Many of these ports are considered “well-known” ports because they appear in an official, published table (RFC 1340) that relates the names of commonly-used protocols with the TCP or UDP ports they typically use.

Table A.1 lists the well-known names recognized by the ASN-9000, and provides the port number associated with each well-known name. Enter the “well-known” port name or number exactly as shown in the table.

Table A.1 - Well Known Names and Ports

Well-known Name	Port Number	Well-known Name	Port Number
at-echo	204	at-nbp	202
at-rtmp	201	at-zis	206
auth	113	bgp	179
biff	512	bootpc	68
bootps	67	chargen	19
courier	530	csnet-ns	105
daytime	13	discard	9

Table A.1 - Well Known Names and Ports

Well-known Name	Port Number	Well-known Name	Port Number
dls	197	domain	53
echo	7	exec	512
finger	79	ftp	21
ftp-data	20	hostname	101
hostnames	101	ingreslock	1524
ipcserver	600	ipx	213
iso-tp0	146	isop-tsap	102
kerberos	88	klongin	543
kshell	544	link	87
login	513	lpd	515
monitor	561	nameserver	42
netbios-dgm	138	netbios-ns	137
netbios-ssn	139	netwews	532
netstat	15	news	144
NeWs	144	new-rwho	550
nicname	43	nntp	119
npp	92	ntp	123
pcserver	600	pop-2	109
pop3	110	printer	515
print-srv	170	rip	520
rlogin	513	rmonitor	560
route	520	rtelnet	107
rwho	513	shell	514
smtp	25	snmp	161
snmptrap	162	sunrpc	111
supdup	95	syslog	514
systat	11	tacnews	98

Table A.1 - Well Known Names and Ports

Well-known Name	Port Number	Well-known Name	Port Number
talk	517	tcpmux	1
telnet	23	tftp	69
time	37	timed	525
uucp	540	who	513
whois	43	x400	103
x400-snd	104	xdmcp	177
supdup	95	syslog	514

Index